

Ref. 1

(19)日本国特許庁 (J P)

(12) 特 許 公 報 (B 2)

(11)特許出願公告番号

特公平6-66051

(24) (44)公告日 平成6年(1994)8月24日

(51)Int.Cl. ⁸	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 9/26	3 1 0	7230-5B		
9/22	3 6 0	7230-5B		

発明の数 2 (全 26 頁)

(21)出願番号 特願昭58-229034

(22)出願日 昭和58年(1983)12月3日

(65)公開番号 特開昭59-111542

(43)公開日 昭和59年(1984)6月27日

(31)優先権主張番号 4 4 6 7 4 8

(32)優先日 1982年12月3日

(33)優先権主張国 米国 (U S)

審判番号 昭63-16572

(71)出願人 999999999

ハネイウエル・インフォメーション・シス
テムス・インコーポレーテッド
アメリカ合衆国ミネソタ州55408, ミネア
ポリス, ハネイウエル・プラザ (番地な
し)

(72)発明者 ミン・ティー・ミュー

アメリカ合衆国マサチューセッツ州01824
チエルムスフォード・チエストナット・ヒ
ル・ロード5

(74)代理人 弁理士 湯浅 恭三 (外3名)

審判の合議体

審判長 松尾 浩太郎

審判官 大橋 隆夫

審判官 岩崎 伸二

最終頁に続く

(54)【発明の名称】 プログラム・カウンタ・スタック方法およびネストされたサブルーチンと命令用の装置

1

【特許請求の範囲】

【請求項1】 マイクロプログラム化された制御記憶装置
によって制御されるデータ処理装置において、前記制御
記憶装置はその記憶ユニットのアドレス可能記憶場所に
保持されたマイクロ命令の異なるシーケンスを実行し、
前記処理装置は前記記憶ユニットの複数のアドレスを保
持するためのプッシュダウスタックを設けられ、前記
データ処理装置を制御して前記マイクロ命令のシーケ
ンスから混合されたサブルーチン分岐と割込みサービ
ス・ルーチンへのエントリを可能にする装置を設け、該装置
が、
実行中のルーチン/サブルーチンから特定されたサブ
ルーチンへの移行を示す第1のマイクロ命令の実行中に制
御されて、前記第1のマイクロ命令のアドレスを増分
し、該増分されたアドレスを前記スタックのトップにプ

2

ッシュする装置と、

前記第1のマイクロ命令の実行中に前記装置が割込みに
応答することを禁止する装置と、
その間に割込みが生ずる第2のマイクロ命令の実行中に
制御されて、前記第2のマイクロ命令の実行中に得られ
る次のアドレスを前記スタックのトップにプッシュし、
その最初のアドレスが前記割込みの種類によって決定さ
れるマイクロ命令サービス・ルーチンへオペレーション
を移行する装置と、

10 各サブルーチン及び前記サービス・ルーチン内の最後の
マイクロ命令によって制御されて、前記スタックをポッ
プアップし、前記スタックのトップから受け取ったアド
レスを戻りアドレスとして用いて前記サブルーチン又は
サービス・ルーチンへ入った前記ルーチン/サブルーチ
ンに復帰する装置と、から成るシステム。

【請求項2】サブルーチン内の前記最後のマイクロ命令の実行と、割込みの発生の同時生起によって可能化されて前記スタックのポップアップを禁止する装置を更に含むことを特徴とする特許請求の範囲第1項記載のシステム。

【請求項3】サブルーチンおよび割込みサービス・ルーチンに対する移行を行う方法であって、

(A) メモリー (24) から、サブルーチンへの分岐を識別するルーチンの第1のマイクロ命令を生成し、

(B) サブルーチンへの分岐を実行し、この分岐実行ステップは、

(a) 禁止可能な割込みの発生を禁止し、

(b) 現アドレス・レジスタ (66) における前記第1のマイクロ命令の現在のアドレスを増分装置 (64) に対して転送し、

(c) 前記現在のアドレスを増分し増分アドレスを形成し、

(d) 戻りアドレスである前記増分アドレスをプッシュ・ダウン・スタック (70) の第1のレジスタ (71) に格納し、

(e) 前記増分アドレスの格納と同時に前記プッシュ・ダウン・スタックにおいて1レベルだけ下の隣接するレジスタ (72, 73, 74) に対して以前の戻りアドレスを転送し、

(f) もし中断状態の割込みが存在しないか、あるいは全ての中断状態の割込みが禁止されているならば、前記第1のマイクロ命令における次のアドレスを次アドレス・レジスタ (63) に対して転送し、

(g) 前記次アドレス・レジスタ (63) における前記次のアドレスを前記現アドレス・レジスタ (66) に対して転送し、

(h) 前記次アドレス・レジスタ (63) からの前記次のアドレスを用いて、前記メモリー (24) からサブルーチンの最初のマイクロ命令である第2のマイクロ命令を生成することからなり、

(C) 割込みサービス・ルーチンへの割込みを生じ、此の割込み生成ステップは、

(a) もし中断状態の禁止されない割込みが存在するならば、最も高い優先順位の禁止されない中断状態の割込みのベクトル化された割込みアドレスを生成し、

(b) もし割込みが禁止されなければ、前記のベクトル化された割込みアドレスを前記次アドレス・レジスタ (63) に対して転送し、

(c) もし割込みが禁止されなければ、前記プッシュ・ダウン・スタックの前記第1のレジスタ (71) に対して前記戻りアドレスである現在のマイクロ命令からの次のアドレスを格納し、

(d) 前記戻りアドレスの格納と同時に前記プッシュ・ダウン・スタック (70) において1レベルだけ下の隣

接するレジスタ (72, 73, 74) に対して以前の戻りアドレスを転送し、

(e) 前記現アドレス・レジスタ (66) に対して前記次アドレス・レジスタ (63) における前記ベクトル化された割込みアドレスを転送し、

(f) 前記次アドレス・レジスタ (63) からの前記ベクトル化された割込みアドレスを用いて前記メモリー (24) から第1の割込みサービス・ルーチンの最初のマイクロ命令である第3のマイクロ命令を生成することを含み、

(D) 前記メモリー (24) から、前記サブルーチンまたは前記割込みからの戻りを識別するマイクロ命令を生成し、

(E) 前記サブルーチンまたは割込みからの戻りを実行し、この戻り実行ステップは、

(a) もし中断状態の割込みが存在しないか、あるいは全ての中断状態の割込みが禁止されているならば、前記プッシュ・ダウン・スタック (70) における前記第1のレジスタ (71) から前記次アドレス・レジスタ (63) に対して前記戻りアドレスを転送し、

(b) 前記次アドレス・レジスタ (63) に対する前記戻りアドレスの前記転送と同時に前記プッシュ・ダウン・スタック (70) において1レベルだけ上げて以前の戻りアドレスを前記隣接するレジスタ (71, 72, 73) に対して転送することを含むことを特徴とする方法。

【発明の詳細な説明】

本発明は、データ処理システムに関し、特にネストされたサブルーチンおよび割込みの実行を行うデータ処理システムに関する。

データ処理システムにおいて実施される諸動作およびこれらの動作が生じる順序は、メモリーに格納されたプログラムにより記述される。実施されるべき一連の操作の各々を完全に規定するというプログラマーの負担を軽減するため、またデータ・プロセッサの能力を更に十分に利用するため、1つのプログラムにおいて複数回要求されたり、あるいは他のプログラムにおいて使用することができる共通の関数が通常提供される。各々の共通関数は、あるシーケンスの命令を含むサブルーチンとして指定することができる。

サブルーチンは、多くの共通関数、例えば数学的近似操作を用いる三角関数に対して使用することができる。一連の操作を何回も書く代わりに、プログラマーはその操作を実行するただ1組の命令を書いて、これを何回も呼び出す。こうして、余弦 (コサイン) のような三角関数の値を必要とする1つのルーチンの場合には、デジタル・データ・プロセッサにおいて余弦サブルーチンに対する1つの分岐操作が実行される。

しばしば、1つのサブルーチンがその動作を完了するために別のサブルーチンを呼び出すことがある。このような状態が生じると、戻りアドレス、即ち分岐サブルーチン

ンからの次のアドレスがデータ・プロセッサにより記憶されて再び呼び出されなければならないので、処理が複雑化する。このように、全プログラムの効率はデータ・プロセッサにおける唯一の場所に特定の1つの関数を提供することによって強化されるが、時間的な配慮およびハードウェアおよび（または）空間の制約の両方の観点から種々のサブルーチンの呼び出しおよびこれからの戻しにおいて効率の低下が生じることがしばしば認められてきた。

従来技術において、サブルーチンに対する移行のための多くの解決策が提供されている。1つの解決法は、サブルーチンの最初の場所への次の命令場所の移動を含むものである。このような構成においては、最後のサブルーチン命令はこのサブルーチンの最初の場所をアクセスする。この最初の場所は、次の命令（即ち、このサブルーチンに分岐した命令の次の命令）に対する戻しアドレスを含み、こうしてデータ・プロセッサがプログラムの順次実施の継続を可能にする。この方法は、いくつかの移行の取扱いが状況を複雑化させるため不利となる。例えば、第1のサブルーチンからこの第1のサブルーチンを使用する第2のサブルーチンへ諸動作を移行することがしばしば有利である。他の状態においては、第1のサブルーチンがそれ自体を再び呼出す場合に有利となることもある。このような移行は問題があり、しばしば修正を行なうことなくあるいは命令数を増加することなく前述の形式のデータ処理システムを用い達成することは不可能である。第1のサブルーチンが最初に呼出される時、主ルーチンにおける次のマイクロ命令のアドレスは最初のサブルーチンの場所に対して転送される。この最初のサブルーチンが中間のルーチンによって再び呼出される時、この中間のルーチンにおける次の命令のアドレスは最初のサブルーチンにおける同じ場所に移動され、このため元の内容を破壊することになる。その結果、最初のサブルーチンは中間のルーチンに戻ることはできるが、主ルーチンへは戻るができない。

第1または第2のサブルーチンが最初のサブルーチンを再び呼出すことを可能にするデータ処理システムにおいては、戻りアドレスを特定の記憶場所へ移動させるため1つの命令が提供される。最後のサブルーチンの命令は、この時、指定された記憶場所のアドレスを含むように変更される。このシステムは1つのサブルーチンが別のサブルーチンを呼出すこと、即ち後者のサブルーチンをネストすることを可能にし、また部分的に完了したサブルーチンが他の目的のため順次使用されることを可能にするが、各ネスト・レベルに対して1つの予約記憶場所およびいくつかの命令が必要となる。ネストされた各サブルーチンに対するこれらの記憶場所の数が増加する程、制御回路の複雑さは増す。更に、最後のサブルーチン命令が各サブルーチン毎に適性な記憶場所をアドレス指定するため修正されなければならないため、プログラ

ミングの複雑さはその度を増す。従って、この試みは、ネスティング・レベル数が増加するに伴い益々厄介な状態となる。

別の形式のデータ処理システムは、前に実行されたサブルーチンまたはルーチンの戻りアドレスを記憶するためにメモリーにおける空白場所を使用する。サブルーチンの最後の命令は指定された記憶場所に対して分岐する。しかし、このシステムを実現するためには、一方が空白の記憶場所に対するポインタであり他方がレジスタのその時の値およびメモリー・アドレスを提供する2つのレジスタが必要となる。この形式のデータ処理装置は前述の問題のいくつかを克服するが、各々の新たなサブルーチンのレベルを表示するための、同時の比較的長い実行時間が必要な追加の命令を依然としてもたらす結果となる。

この問題の更に最近の解決法は、参考のため本文に引用される米国特許第3,909,797号「ネストされたサブルーチンに対する制御ストア制御および後入れ先出しスタックを用いるデータ処理システム」において見出される。本米国特許においては、データ処理システムは中央処理サブシステムの順次動作を可能にするマイクロプログラム制御ストア装置を有する。本特許は、制御ストア装置を制御するファームウェア内でプログラムされたサブルーチンのネスティング方法について記載しているが、この発明の原理はソフトウェア・プログラムまたはファームウェア・マイクロプログラムに対しても等しく適用可能である。制御ストア制御は、その時作動中のマイクロプログラム・ルーチンの戻りアドレスの記憶を可能にするためのマイクロ指令を介してマイクロプログラム・サブルーチンに対する分岐マイクロ命令を提供する。このマイクロ指令はまた、以前に記憶された戻りマイクロプログラム・アドレスが1レベルだけ下方に移動（プッシュ）されるように後入れ先出しスタックを可能状態にする。マイクロプログラム・サブルーチンの完了と同時に、制御ストア装置は、マルチプレクサが後入れ先出しスタックの最上位に含まれる戻りアドレスを選定することを可能にし、かつこれをシステムの現在アドレス・レジスタに対して提供するための分岐フィールドを提供する。この分岐フィールドまたは、後入れ先出しスタックにおける前に記憶された戻りアドレスを1レベルだけ上げる。

プログラムの実行に対して非同期的に生じる事象にตอบสนองさせるため、データ処理システムにおいて割込みが使用される。これらの非同期的事象は、データ処理システム内の障害、もしくはデータ処理システムにより制御される諸装置の状態の変化である。例えば、データ処理システムの割込みによって、入出力（I/O）装置は、これがデータ処理システムによるアテンションまたはサービスを要求することを信号することができる。このような割込みは、I/O操作の完了またはメモリーのエラーの

ようなシステムの障害の発生を信号するため使用することもできる。サブルーチンの場合のように、割込みはプログラムのフローを転向させることができる。割込みの場合には、プログラムのフローは、割込みを生じた事象のタイプを取扱うようにプログラムされた割込みサービス・ルーチンに対して転向される。割込みサービス・ルーチンの実行の完了と同時に、割込みサービス・ルーチンは、データ・プロセッサが割込みと同時に記憶した戻りアドレスにおいてプログラムの実行を再開させねばならない。

戻りアドレスを記憶するための1つの方法は、メモリー内の固定場所を割込みされたプログラムの戻りアドレスを記憶するために各タイプの割込み毎に割当てることである。この方法を用いて、割込みサービス・ルーチンが完了した時、ある特定の形式の割込みと関連する固定された記憶場所からデータ・プロセッサのプログラム・カウンタを再ロードすることができ、その割込みの発生の直前に実行中であったプログラムの実行を再開することができる。この方法は、割込みが行なわれたプログラムに戻すためにこのサービス・ルーチンが各割込み毎に異なる固定された記憶場所から割込み戻りアドレスを検索しなければならない故に、割込みサービス・ルーチンが容易に共通のプログラム命令を共用できないという短所を有する。

割込み戻りアドレスの取扱いに関する異なる試みは、参考のため本文に引用された米国特許第4,340,933号「存在しないメモリー・アドレスの集中検出法を有するデータ処理システム」において見出される。ファームウェア・マイクロプログラムの実行に割込みを行なうハードウェア割込みを取扱う本米国特許においては、マイクロプログラムの戻りアドレスはある特殊なレジスタ（ハードウェア割込み戻りアドレス・レジスタ）に記憶され、又更に別のハードウェア割込みはマイクロプログラム・ハードウェア割込みサービス・ルーチンの完了まで阻止される。以降の割込みの禁止は、前記のハードウェア割込み戻りアドレス・レジスタに記憶された最初のマイクロプログラムの戻りアドレスが第2のハードウェア割込みの発生によりその上に記憶されようとする第2の戻りアドレスにより破壊されないことを保証するものである。この方法は、1つの割込み戻りアドレス・レジスタからプログラム・カウンタを再ロードすることにより全ての割込みサービス・ルーチンが割込みが行なわれたプログラムに対する実行を再開するので、割込みサービス・ルーチンが共通のプログラム命令を共用することを可能にするが、依然として割込みおよびサブルーチンが容易に共通のプログラム命令を共用できないという短所を有するものである。

従って、必要とされるものは、共通かつ有効な方法においてサブルーチン呼出しのネスティングおよび割込みのサービス操作を可能にする戻りアドレスの記憶方法であ

る。

従って、本発明の主な目的は、サブルーチンへの移行および割込みが迅速化される改善されたデータ処理システムの提供にある。

本発明の別の目的は、サブルーチンへの分岐または非同期割込みに対する応答の際戻りアドレスを記憶し、また完了と同時に分岐されたサブルーチンまたは割込み応答ルーチンが戻りアドレスにおけるプログラムの実行の再開を自動的に可能にするための新規な改善されたシステムの提供にある。

本発明の他の目的は、ネストされたサブルーチンや割込みを利用するためのデータ処理および計算システムにおいて使用される、効率が良く、信頼性が高く、全体的な時間の節減をもたらす改善された手法の提供にある。

本発明は、特に頭書の特許請求の範囲において規定されるものである。本発明の上記および他の目的および利点については、図面に関して以下の記述を照合することにより理解が得られよう。

前記の諸目的は、データ処理システムのマイクロプロセッサにおいてこのマイクロプロセッサの順次操作を可能にする制御領域を提供することにより、その1つの操作モードに従って本発明の一実施態様によって達成される。この制御領域は、その時作動中のルーチンの戻りアドレスを記憶することを可能にするためのマイクロ指令を介して前記サブルーチンに対する分岐マイクロ命令を提供する。前記マイクロ指令はまた、前に記憶された戻りアドレスが1レベルだけ下方にプッシュされるように、後入れ先出しスタックを可能状態にする。前記制御領域は更に、最高優先順位の禁止状態にない中断状態の割込みと対応するベクトル化された割込みアドレスの生成を行なう。これらのベクトル化された割込みアドレスは、その時のマイクロ命令により生成されたアドレスの代りに次アドレス・レジスタに対して与えられ、その時のマイクロ命令により生成されたアドレスは割込みの発生と同時に1レベルだけ後入れ先出しスタックに対してプッシュされる。サブルーチンもしくは割込みサービス・ルーチンの完了と同時に、制御領域は、マルチプレクサが制御領域における次のアドレス・レジスタに対して与えられるべき後入れ先出しスタックの最上位に含まれる戻りアドレスを選択することを可能にするためのアドレス・フィールドを提供する。前記後入れ先出しスタックの取り出し操作はまた、前に記憶された戻りアドレスを1レベルだけ上昇させる。サブルーチンまたは割込みサービス・ルーチンから戻る時もし割込みが生じるならば、後入れ先出しスタックが取り出し操作もしくはプッシュ操作のいずれも行なわれないように、またベクトル化されたアドレスが実行されるべき次のマイクロ命令のアドレスとして次アドレス・レジスタに対して与えられるようにする装置もまた設けられる。望ましい実施態様においては、後入れ先出しスタックはシフト・レジスタ

を用いて構成されている。

本発明のプロセスが実施される方法、および本発明の装置が構成される方法およびその操作モードについては、同じ照合番号がいくつかの図における類似の要素を識別する図面と共に、以下の詳細な記述に照合すれば最もよく理解することができよう。

本発明の背景の前記の論述は主としてソフトウェアおよびソフトウェア・ルーチンおよび割込みのネスティングに関するものであり、その原理はファームウェア・ルーチンおよび割込みに対して等しく適用することができる。本発明の以下の議論は、本発明がファームウェア・サブルーチンおよび割込みのネスティングにおいて使用するためマイクロプロセッサ内部で実施される望ましい実施態様に関するものである。

図面に関連して本発明の構造および作用についての記述に先立つて、説明の便のため、図面中に示される実施態様は4つのサブルーチンまたは割込みレベルのネスティングが可能であるように示されていることに注意されたい。しかし、無論、実施においては、使用されるハードウェアを適当に増加することにより4つより遥かに大きくすることもできることを理解されよう。あるいはまた、ハードウェアの拡張を行なう代りに、桁溢れ手法を用いることもできる。この場合は、4つ以上のサブルーチンまたは割込みレベルのネスティングを欲する場合においては、マイクロプロセッサは4つを越えるサブルーチンまたは割込みの取扱いのため他の構成の機構によることもできる。

図面に示された実施態様の構成は、ある回路構成において有効である。しかし、本発明を種々の形式のデータ・プロセッサに適用するため本発明の実施において他の論理構成を使用することも可能であることが理解されよう。従って、本発明は図面に示された特定の構成に限定することを意図するものではない。

次に図面によれば、第1図は、本発明のネストされたサブルーチンおよび割込み構成を用いることができるシステム全体を示している。特に、第1図は、主記憶装置サブシステム10と、中央演算処理装置(CPU)20と、入出力(I/O)装置40とを示している。主記憶装置サブシステム10は、3個の金属酸化物半導体モジュール12、14、16からなる。この3個のモジュールは、中央演算処理装置20と入出力装置40に対して主要バス26を介してインターフェースされている。主要バス26は、全てのメモリ・モジュールおよび入出力装置に対するアクセスおよび制御を提供する能力を与える。

中央演算処理装置20は、固定された長さのフィールドおよび変更可能な長さのフィールドについて操作するワード指向の命令を実行する。中央演算処理装置における情報の基本単位は、2つの8ビットのバイトからなる16ビット・ワードである。これら情報ワードは、命令、または固定小数点もしくは浮動小数点のオペランドに対して

1つ、2つまたは4つのグループにおいて使用される。バイトまたは10進数または英数字のデータとして可変長さフィールドにおいても使用される。

CPU20は、マイクロプロセッサ30と、モニター・ロジック22と、読出し専用記録装置(ROS)24とからなっている。マイクロプロセッサ30は、望ましい実施態様においてROS24に含まれる48ビットの外部ファームウェアのマイクロ命令ワードにより駆動される、算術演算、論理演算および制御操作が可能な16ビットのNMOSチップである。マイクロプロセッサ30の構成は、CPU20の16ビットのソフトウェア命令のレパートリの実行を可能にする。マイクロプロセッサ30は、全体のシステム設計を容易にするため入出力(I/O)および記憶操作を直接制御するように構成されている。マイクロプロセッサ30の構成は、48ビットのマイクロ命令ワード当り12個までの同時のマイクロ演算を可能にする真の水平マイクロプログラミングを行なう48ビットの外部のファームウェアのマイクロ命令の使用によって比較的大きな制御および統合を可能にする。マイクロプロセッサ30の構成はまた、ファームウェア・マイクロプログラム・ルーチンに対するベクトルの生成を行なう8つの外部のハードウェア割込みを可能にすると共に、ファームウェアの制御下で取扱われる5つの外部ソフトウェア割込みを可能にする。更に、マイクロプロセッサ30は、モニター・ロジック22において生成され、ファームウェア内部で実施されるべき複雑な分岐操作を可能にするマイクロプロセッサの制御域36内のロジックによる複雑な検査分岐および主要分岐操作によって検出され制御される10個の外部モニター・ビットを提供する。

マイクロプロセッサ30は、第1図に示されたような5つの主な内部のハードウェアの論理領域からなる。この5つの主な論理領域は、演算論理装置(ALU)を含むデータ操作領域32、メモリー管理装置(MMU)34、制御領域36、プロセッサ・バス37および内部バス38である。

プロセッサ・バス37は、20本のアドレス/データ回線、1つのメモリー・アドレス干渉回線および3つの汎用制御回線からなっている。プロセッサ・バス37は主要バス26と接続され、また主記憶装置10および入出力装置40に対してアドレスを提供しかつ主記憶装置10および入出力装置40に関してデータを送受するために使用される。

内部バス38は、マイクロプロセッサ・チップの他の4つの領域間の通信のための主な経路である。内部バス38は20ビット巾である。48ビットのマイクロ命令ワード内の11個のマイクロopの制御下で内部バス38に対する情報の12の提供源が存在する。ALUは、11個の指定されたマイクロopがいずれも使用されなければ内部バス38に対する省略時のソースとなる。

データ操作領域32は、データに関する算術演算および論理演算を行ない、メモリー・アドレス生成を行なう。データ操作領域32は、1つの標識レジスタ、種々の個別的

10

20

30

40

50

レジスタ、レジスタ・ファイルおよび演算論理装置 (ALU) およびシフト機構からなっている。これらの個別的レジスタの1つは、CPUに対する20ビットのメモリー・アドレス・レジスタとして使用される。このプログラム・カウンタは、中央演算処理装置20により実行中のソフトウェア命令の20ビットのメモリー・アドレスを保有する。レジスタ・ファイルは、7つの16ビットのソフトウェア・アドレス指定可能なデータ・アドレス・レジスタ、7つのソフトウェア・アドレス指定可能な基底レジスタ、およびその全てがCPU20により実行されつつあるソフトウェア・プログラムと関連する他の種々のレジスタを含む。このレジスタ・ファイルまたは、ファームウェア操作中情報を一時的に記憶するための、ROS24からのファームウェア・プログラムによって使用される種々の作業レジスタを保有する。シフト機構は、16ビットまたは32ビットのオペランドに関して種々のシフト操作 (即ち、開閉、演算論理、左側右側) を行なう能力に使用される。

マイクロプロセッサ30の制御領域36は、3つの領域、即ち、制御のための入力ラッチ、検査可能なレジスタおよび次のアドレス生成の領域に論理的に分割されている。制御領域36については、第2図に関して以下に更に詳細に記述する。

マイクロプロセッサ30のMMU34の部分は、主として下記のものからなっている。即ち、1つのレジスタ・ファイル、ベース再配置用の12ビット・アドレス、メモリー・セグメントの大きさを検査する1つの9ビットコンパレータ、あるセグメントに対するアクセス権を評価するいくつかの2ビットのリング・コンパレータ、および潜在的なメモリーの干渉を表示する記憶フリップフロップである。CPUが生成したメモリー・アドレス・サイクルの間、MMU34は、内部バス38により与えられるセグメント番号、ブロック番号およびオフセット値を含むソフトウェアの論理アドレスをプロセッサ・バス37にみちびかれ、更に主要バス26を介して主記憶装置10に対して与えられる物理的地址に翻訳する。

これまでの記述により理解できるように、CPU20は、その命令が主記憶装置10から取出されるソフトウェア・プログラムを実行し、これもまた主記憶装置10に保有されるデータに関する演算処理操作を実施する。CPU20により実行されるソフトウェア・プログラムは、ソフトウェア・ビジブルである汎用レジスタおよび基底レジスタを操作する能力を有し、その時のソフトウェア命令はプログラム・カウンタにより指示される。CPU20により実行中のソフトウェアからビジブルであるこれらの汎用レジスタ、基底アドレス・レジスタおよびプログラム・カウンタは、マイクロプロセッサ30のデータ操作領域32内に物理的に保有される。

第1図のCPU20の詳細な作用は、読出し専用記憶装置 (ROS) 24に記憶されたファームウェアのマイクロ命令の制

御下でマイクロプロセッサ30によって制御される。ROS24における各記憶場所はマイクロプロセッサの1つの機械サイクルを制御するように解釈することができる。ROS24の各場所が読出される時、その内容は制御領域36によって復号され、その結果マイクロプロセッサ30内部で特殊な操作を惹起する。ROSの場所をグループ化することにより、CPU20と関連する特殊な操作即ちソフトウェア命令を実施することができるファームウェア・マイクロ命令シーケンスが得られる。各ソフトウェア命令が開始されると、ソフトウェア命令の命令コード・フィールド内のあるビットが用いられてROS24内に保有されるファームウェア・マイクロ命令ルーチンの始動アドレスを判定する。マイクロプロセッサ30により行なわれるソフトウェアの命令復号によりセットまたはリセットされるあるフリップフロップのテストは、必要に応じてマイクロプロセッサがROS24内部の更に特定のファームウェア・マイクロ命令シーケンスに対して分岐することを許容する。

主要バス26に対しては、入出力装置40が接続されている。入出力コントローラ42は、周辺装置44から主要バス26を介して主記憶装置10に対するデータ経路を完成する入出力装置の部分である。I/Oコントローラ42は、データ転送をもたらす結果となる制御に加えて周辺装置の指令が開始される経路を提供する。

次に第1図の制御領域36を更に詳細に示す第2図においては、制御領域36は別のロジックおよび回路を含むが、本発明の目的のためにはこのロジックは第2図に示されるものに限定されている。第2図はまた、内部バス38と、モニター・ロジック22と、読出し専用記憶装置 (ROS) 24を示している。

第2図においては、ブロックの右上隅部に隣接する数字は、ブロックにより表わされるレジスタに含まれる情報のビット数を表わす。

ROS24は、ファームウェアのマイクロ命令を保持することができる読出し専用メモリー (ROS)、またはランダム・アクセス・メモリー (RAM) または他のどんな形態の記憶素子でもよい。ROS24は、中央演算処理装置の動作を制御するため、特にCPU20のソフトウェア命令を実行するためマイクロプロセッサ30により使用されるファームウェア・マイクロ命令 (即ち、制御ワード) を保有する。マイクロプロセッサの機械サイクル毎に、制御ワードがROS24から取出される。

ROS24は、この読出し専用記憶装置24から取出されるマイクロ命令ワードを受取るROSデータ・レジスタ65に対して接続されている。各マイクロ命令は、アドレス部分および指令部分を有する。マイクロ命令ワードにおけるアドレス部分は、マイクロプロセッサ30により実行されるべき次のマイクロ命令となる読出し専用記憶装置24から読出される次の場所のアドレスを識別する。マイクロ命令の指令部分は、現在のマイクロ命令の実行の間マイ

クロプロセッサにより実施される操作を識別する。マイクロ命令ワードのアドレス部分は、例えば望ましい実施態様においてはマイクロ命令ワードのビット0乃至12（第3図参照）に保持される予め定められた数のビットに保持することができる。マイクロ命令の指数部分もまた、例えば望ましい実施態様においてはマイクロ命令ワードのビット13乃至47（第3図参照）に保持される予め定められた数のビットに保持することができる。この指令部分は、更に、マイクロ命令の副指令を構成するいくつかのフィールドに分割することができる。

第3図に関してマイクロ命令ワードを更に詳細に記述する前に、第2図の他の要素について記述する。モニター・ロジック22はCPU20に関する状態情報を提供し、この状態がファームウェアによりテストできるようにテスト・フリップフロップ50にロードされる。モニター・ロジック22からの動的状態の情報の10ビットの保持に加えて、テスト・フリップフロップ50は種々のCPUの選択状態をサンプルする5つのビットを保持する。このCPU選択ビットは、性格において静的と考えるべきで、特定のハードウェアの選択がデータ処理システム内部に存在するかどうかを表示する。更に、テスト・フリップフロップ50はファームウェアの制御下でセットまたはリセットされ、またはビットを転送させるため使用可能な4つの制御フリップフロップを有する。これらの4つの制御フリップフロップはファームウェアによりテスト可能である。また、ALUのビット16からの桁送りまたは桁溢れがあったかどうか、またはALUのビット20からの桁送りまたは桁溢れがあったかどうか、または内部バス38におけるあるビットが0に等しいかどうか、等の如き動的状態で各ファームウェア・サイクル毎にロードされる10個の一時フリップフロップがテスト・フリップフロップ50に存在する。これら10個の一時フリップフロップもまたファームウェアによってテスト可能である。

Fレジスタ51は、内部バス38からロードされる16ビットの命令レジスタである。Fレジスタの全てのビットはファームウェアによりテスト可能である。Fレジスタ51の下位の4ビットはまた、5ビットのカウンタであるFカウンタ52の下位の4ビットを構成する。Fカウンタ52は、内部バス38からロードすることができる5ビット・カウンタである。Fカウンタ52は増進または減退させることができる。Fカウンタ52の4つの下位のビットはまた、16ビットのマスクをファームウェアの制御下で内部バス38に置くことができるように復号される。

ソフトウェア割込みが生じる5つの可能な条件がある。これらの条件はソフトウェア割込みレジスタ53においてラッチされる。ソフトウェア割込みブリネット54は、これらの条件に優先順位を付して主分岐ロジック57に対する入力用のベクトル化されたアドレスを生成する。

制御領域36の次アドレス生成部分55は、読出し専用記憶装置（ROS）24の順次動作に必要なロジックを保有す

る。テスト分岐ロジック56は、ROSアドレス・レジスタ63に対する2通りの分岐アドレスを生じることができる64個のテスト条件をテストするために使用される。これらの64個のテスト条件は、ファームウェアの制御下でテスト分岐ロジック56のアドレス・マルチプレクサ160に対する情報の1ビットの出力としてテスト可能である。テスト分岐ロジック56に対する入力、テスト・フリップフロップ50、Fレジスタ51およびFカウンタ52によって提供される。

- 10 主分岐ロジック57は、15個の主なテスト分岐マトリックスを与える。これらマトリックスに対する入力の大部分はFレジスタ51からの（種々の組合せにおける）ものである。他の入力は、テスト・フリップフロップ50のモニターおよび選択ビットからのものである。主分岐ロジック57の出力側は、アドレス・マルチプレクサ160に対して与えられるアドレス情報の4ビットである。
- レジスタ58は、ハードウェア割込みを生じる10個の可能な条件と対応する情報のビットを提供する。ハードウェア割込みブリネット59は、これらの10個の可能な条件に優先順位を付して、これら10個の可能な条件の内の1つが生じる時12ビットのベクトル化されたハードウェア割込みアドレスを生じるためアドレス・マルチプレクサ262により使用される4ビットの出力を生じる。
- アドレス・マルチプレクサ160の出力は、ROSアドレス・レジスタ63にロードされてROS24からの次のマイクロ命令を取出すため使用される12ビットの形式的な次アドレスを与える。この12ビットのアドレスは、もしハードウェア割込みが生じなければこの形式的な次アドレスだけが次アドレスとして使用されるという意味において形式的である。もし中断中のハードウェア割込みがないかあるいは中断中のハードウェア割込みがマイクロ命令ワード内の割込み禁止ビットのセットによって禁止されているならば、ハードウェア割込みは生じない。

（第3図におけるビット34参照）

アドレス・マルチプレクサ262は、マルチプレクサ160により生成される12ビットの形式的な次アドレスと、ハードウェア割込みブリネット59からの4ビットを8つの先行する0ビットと組合せることにより形成されるベクトル化ハードウェア割込みアドレスとの間を選択するため使用される。アドレス・マルチプレクサ262の出力は、ROSアドレス・レジスタ（RAR）63に対してロードされる12ビットの次アドレスである。RAR63の出力は、ROS24から取出されるべき次のマイクロ命令のアドレスを提供するために使用される。RAR63の出力はまたROSアドレス履歴レジスタ66に対する入力ともなる。

ROSアドレス履歴レジスタ66は、次マイクロ命令アドレスが生成されてROSアドレス・レジスタ63に対して転送されつつある間ROSデータ・レジスタ65に保有される現在のマイクロ命令の実行の際早期に、ROSアドレス履歴レジスタが現在のマイクロ命令のアドレスを保持するよ

うに設けられている。この現在のマイクロ命令アドレスは、もし現在のマイクロ命令がその使用を要求するならば、次のマイクロ命令アドレスの生成の際に使用される。ROSアドレス履歴レジスタ66からの現在のアドレスはまた、マイクロサブルーチンからの戻りアドレスとして増分機構64で増分された後に使用される。

増分機構64は、ROSアドレス履歴レジスタ66に保有されるアドレスを予め定めた数だけ（例えば、望ましい実施態様においては1だけ）増分する。増分機構64は、PUSHマイクロ指令の間戻りマルチプレクサ61を介して戻りアドレス・スタック70のソースとして使用されることとなる12ビットの増分機構である。増分機構64の出力はまた、アドレス・マルチプレクサ160およびアドレス・マルチプレクサ262を介してINCおよびINCKマイクロ指令のためにRAR63に対して次のROSアドレス値を与えるためにも使用される。このINCマイクロ指令は1だけ増分された現在のROSアドレスが次のROSアドレスとなることを指定し、INCKマイクロ指令は現在のアドレス値プラス1が次のROSアドレス値となるべきこと、およびマイクロ命令内の他の未使用のアドレス・フィールド・ビットにおいて指示される定数が内部バス38に置かれるべきことを指定する。

戻りアドレス・スタック70は、サブルーチンおよびハードウェア割込みの戻りアドレスの格納のために使用される4×12ビットの後入れ先出し（LIFO）アレーである。戻りアドレス・スタック70はクリア期間中16進数001の値に初期化され、その最下部の場所は各ポップ（戻し）マイクロ指令の間001（16進数）にセットされる。PUSHマイクロ指令は、戻りアドレス・スタック70の最上位のソースとして増分機構64の出力を用いる。ハードウェア割込みは、戻りアドレス・スタック70の最上位のソースとして形式的な次アドレスであるアドレス・マルチプレクサ160の出力を用いる。増分機構64は、ROSデータ・レジスタ65の次の指令部分の1つのサブフィールドがPUSHマイクロ指令を指定する時、戻りアドレス・スタック70に対してROSアドレス履歴レジスタ66からの増分されたアドレス履歴を転送する。このPUSHマイクロ指令は、マイクロプログラムのサブルーチンへの分岐の間実行中のマイクロプログラムのマイクロ命令の戻りアドレスの格納を可能にする。

PUSHマイクロ指令に回答して、増分機構64はROSアドレス履歴レジスタ66からの複数のレジスタ71乃至74からなる戻りアドレス・スタック70に対して増分された現在のROSアドレスを転送する。機能的には、戻りアドレス・スタック70は列状に配列された複数の作業レジスタから成る後入れ先出し記憶装置である。このスタックからの唯一の出力は、アドレス・マルチプレクサ160に対して接続された最上部のレジスタ71からである。戻りアドレス・スタック70に対する入力是最上部と最底部からのものだけである。1つのアドレス・がスタック70に対し

てプッシュされる時、これはこのスタックに既にある他のアドレスが1つのレジスタだけ列の下方に下げられた後レジスタ71に行く。1つのアドレスが列から除去（ポップ・アップ）される時、これは最上部のレジスタ71からアドレス・マルチプレクサ160に与えられ、戻りアドレス・スタック70に格納された各アドレスは1つのハードウェア・レジスタだけ列の上方に移動する。このポップ操作の間、空白になる底部のレジスタ74はアドレス001（16進数）でロードされる。このスタックはカードのデッキとして視ることが可能であるが、これにおいてはこのデッキのカードに対するアクセスは一時に1枚ずつデッキの最上部のカードを加除することによってのみ可能であり、また各カードがデッキの最上部から取り出される毎に予め定めたあるカード（16進数値001）がデッキの底部に対して加えられる。

このように、戻りアドレス・スタック70は、実行中のマイクロプログラムがサブルーチンに対して分岐する時ROSアドレス履歴レジスタ66から与えられる増分された現在アドレスを格納する。更に、生じた特定のハードウェア割込みにより決定されるROS24内の予め定めた場所に対してファームウェアの実行を誘導するハードウェア割込みが生じる時は常に、戻りアドレス・スタック70はアドレス・マルチプレクサ160により供給される形式的な次アドレスを格納する。戻りアドレス・スタック70に格納されたこれらのアドレスは、マイクロプログラムサブルーチンへの分岐またはマイクロプログラム割込処理ルーチンへの分岐が発生しなければすぐに実行されたはずの、次のマイクロプログラムのステップを指示する。マイクロプログラムのサブルーチンへの分岐が生じる

時、あるいはハードウェア割込みが生じる時これらのアドレスが格納されるため、戻りアドレス・スタック70におけるアドレスは、あるサブルーチンまたはハードウェア割込み取扱いルーチンにおける最後のマイクロ命令の実行と同時に、このマイクロプログラムを適性なシーケンスに戻すことになる。このことは、以下に記述される如き第4図を照合することにより更によく示されよう。

第3図は、望ましい実施態様のマイクロプロセッサ30のファームウェアのマイクロ命令ワード・フィールドを示している。このマイクロ命令ワードは、48ビット（ビット0～47）からなっている。ビット0乃至12はROSアドレス・フィールドとして使用され、ビット13乃至17はレジスタ・ファイルにおけるレジスタの選択のため使用され、ビット18乃至22はALUの演算論理機能およびそのポートに対する入力の制御のため使用され、ビット23乃至25はバス制御として使用され、ビット26乃至30はレジスタの変更フィールドとして使用され、ビット31乃至33はメモリー管理装置の制御用として使用され、ビット34はハードウェア割込みの発生を禁止するため使用され、ビット35乃至47は特殊な制御フィールドとして使用される。

特殊な制御フィールド (RDDTビット35乃至47) は、マイクロ命令ファームウェア・ワードにおける他のフィールドのあるものの変更および追加のため使用される。特殊制御フィールドは、1つのマイクロサイクルの間に3つまでの同時のマイクロ指令を提供する。この特殊制御フィールドは、第3図に示されるように4つのサブフィールド (A乃至D) に分割される。このサブフィールドのいくつかの解釈は他のサブフィールドの内容に依存している。

前記のマイクロ命令の48ビットは、マイクロ命令の実行の開始においてROSデータ・レジスタ65に対してロードされる。これら48のビットは、信号RDDT00乃至RDDT47と呼ばれる。

ROSアドレス・フィールドは13ビット (RDDT00乃至RDDT12) を含みマイクロプログラム・シーケンスにおける次のファームウェア・ステップのアドレスを生成するため使用される。この次アドレスを生成するための方法は、下表1に示される如くROSアドレス・フィールドの最初の5つのビットにより規定される。即ち、

RDDTビット					動作内容
0	1	2	3	4	
1	X	X	X	X	飛越し
0	1	X	X	X	テスト分岐
0	0	1	X	X	主要分岐
0	0	0	1	X	定数により増分
0	0	0	0	1	定数によらない増分
0	0	0	0	0	戻り (POPマイクロ指令)

マイクロ指令PUSHは、表1にリストされた最初の5つの操作のいずれとも関連して使用することができる。このマイクロ命令PUSHは、飛越しまたは分岐のマイクロ指令と組合せて使用される時、マイクロプログラムが分岐させられたサブルーチンの完了と同時に戻ること欲する戻りアドレスを戻りアドレス・スタック70に対して格納させることを許容する。マイクロプログラムを書く人によるこの戻りアドレスの格納を容易にするため、マイクロ指令PUSHは増分機構64により1だけ増分されるROSアドレス履歴レジスタ66の内容を戻りアドレス・スタック70の最上部にプッシュする。そして戻り (POP) マイクロ指令は、マイクロプログラムのサブルーチンと呼んだマイクロ命令の直後の場所に対して帰る、呼出されたサブルーチンの最後のファームウェアのステップとしてマイクロプログラムにより使用されるのである。

表1において記述される6つの操作により規定される次アドレスの生成に対する1つの例外は、ハードウェア割込みのそれである。ハードウェア割込みが開始されると、次のROSアドレスが1つのハードウェア・ベクトルとして与えられ、現在のファームウェア・ワードのROSアドレス・フィールドにより発生されるROSアドレス

が、戻りマルチプレクサ61の出力として選択されて戻りアドレス・スタック70に対してプッシュされるアドレス・マルチプレクサ160の出力により戻りアドレス・スタック70の最上部に置かれることになる。もし表1に示される最初の5つの操作の内の1つと関連してマイクロ指令PUSH (サブフィールドAおよびBの特殊なコーディングと関連してビットRDDT42乃至RDDT44における特殊な制御フィールドCにより規定される如き) が使用されるならば、マイクロプログラムはまた、ハードウェア割込みの発生の結果として実施される競合するプッシュ操作を阻止するため、RDDTビット34を2進数1にすることによってハードウェア割込みを禁止しなければならない。ファームウェアのマイクロプログラム・シーケンスの完了に先立って何時でもハードウェア割込みが (通常、外部の非同期事象の発生により) 生じ得るため、ハードウェア割込みの発生を許容するため特別な配慮が取られなければならない。

表1に規定された6つの操作の分岐能力はページ分岐およびバンク分岐操作と呼ばれる。ページはROS24内部の64個の連続する記憶場所として定義され、バンクはROS24内の1024個の記憶場所 (16ページ) として定義される。テスト分岐操作に対する分岐の境界は、1つのページ内のいずれかの場所に限定されている。主要分岐のための分岐境界は1つのバンク内の任意の場所に限定される。表1の残る4操作は、1つのバンクから他のバンクへの分岐または増分操作が可能である。

飛越し操作は、ROS24の可能な4096の場所のどこかに対する分岐を可能にする表1の唯一の次アドレス生成法である。これは、ROSアドレス・フィールド内で、実行されるべき次のファームウェア・マイクロ命令の12ビットの直接アドレスを提供することにより達成されるのである。

RDDTビット0が2進数1と等しい時、ROSデータ・レジスタ65に保持される如きその時のROSデータ・ワード (ファームウェアのマイクロ命令) のRDDTビット1乃至12は、無論介入するハードウェア割込みが生じないことを仮定して、ファームウェアのマイクロプログラムにおける次のアドレスとして、アドレス・マルチプレクサ160とアドレス・マルチプレクサ262を介してROSアドレス・レジスタ63に対し直接送られる。ハードウェア割込みが生じる場合は、この形式的な次アドレスは戻りアドレス・スタック70の最上部に対してプッシュされ、ハードウェア割込みプリネット59の4ビット出力に対して8つの上位の2進数0のビットを連結することにより生成されるハードウェア割込みベクトル・アドレスがアドレス・マルチプレクサ262により出力される次のROSアドレスとしてROSアドレス・レジスタ63に対してロードされることになる。

マイクロ指令PUSH (特別の制御フィールドCにより定義される如き) は飛越し操作と共に使用することができ

る。ハードウェア割込みの禁止ビットRDDT34は、マイクロ指令PUSHが飛越し操作と関連して使用される時、ハードウェア割込みを禁止するため2進数1にセットされなければならない。マイクロ指令PUSHの間、ROSアドレス履歴レジスタ66に保持されている現在のROSアドレスは1だけ増分され、RDDTビット1乃至12に指定される如き次のアドレスがROSアドレス・レジスタ63に対してロードされる前に、戻りマルチプレクサ61によって戻りアドレス・スタック70の最上部に対してプッシュされる。

表1のテスト分岐操作は、ビット3乃至8においてROSアドレス・フィールドの一部として指定される64のテスト条件の1つの結果を用いる2通りの分岐である。全てのテスト分岐は、その時のページ内の分岐に限定される、即ちテストの結果として生成される次のROSアドレスは常に(テストの結果、即ち真か偽であるかに従って)8つの場所だけ離れているがROSアドレス・レジスタ63による現在のアドレス(64の場所)内にある2つの場所の内の1つとなる。即ち、テスト分岐の結果に従って、形式的な次のROSアドレス内のビット位置7、8、10、11および12に対してそれぞれ使用されるROSアドレス・フィールドのビット2、9、10、11および12からの6つの下位のビットと(RoSアドレス履歴レジスタ66からの)現在のROSアドレスから6つの上位ビットとを連結したものにより決定された場所からの次のマイクロ命令が取出されることになり、テストの結果(それぞれ真または偽と対応する1または0)が形式的な次のROSアドレス内のビット位置9として使用される。

現在のファームウェア・マイクロ命令ワードにおけるビット0乃至13で特定されるテスト分岐の結果として生成された前述の如きビットからなる形式的な次のROSアドレスは、介入するハードウェア割込みが生じなければ次のROSアドレスとなる。ハードウェアの割込みが生じれば、このアドレスは戻りアドレス・スタック70の最上部に置かれ、生成されたハードウェア割込みベクトル・アドレスがこれをROSアドレス・レジスタ63における次のROSアドレスの内容として置換する。

マイクロ指令PUSHはまたテスト分岐操作と共に使用することができる。前述の如く、ハードウェア割込みは、マイクロ指令PUSHが用いられるならば(RDDTビット34を2進数1に等しくセットすることにより)禁止されなければならない。もしマイクロ指令PUSHがテスト分岐操作と関連して用いられるならば、(ROSアドレス履歴レジスタ66からの)テスト分岐マイクロ命令のアドレスであるその時のROSアドレスを1増分したものが戻りアドレス・スタック70の最上部に置かれ、マイクロ指令はテスト分岐ロジック56の出力により決定される形式的な次のアドレスに対して分岐することになる。

主要分岐操作は、ビット5乃至8におけるROSアドレス・フィールドの一部として指定される15のテスト・グループの結果を用いる16通りの分岐である。全ての主要分

岐はその時のバンク(1024の場所)内の分岐に限定される。即ち、主要分岐テストの結果として生成される形式的な次のROSアドレスは(主要分岐マトリックスの出力に従って)常に16の場所の内の1つにあり、離散状態であるがバンク(1024の場所)内にある16の場所はその時ROSアドレス履歴レジスタ66によりアドレス指定される。

形式的な次のROSアドレスは現在のROSアドレスからビット0および1を取り、これらを形式的な次のROSアドレスにおけるビット0および1として用い、またROSアドレス・フィールドからビット3、4、9、10、11および12を取りこれらをそれぞれ形式的な次のROSアドレスにおけるビット2、3、8、9、10および11として用いることによって生成される。更に、形式的な次のROSアドレスにおけるビット14乃至7は、主要分岐ロジック57による4ビット出力により決定される。

前述の如くROSデータ・レジスタ65に保有される現在のファームウェアマイクロ命令ワードのビット0乃至12に指定された主要分岐操作の結果として生成される形式的な次のROSアドレスは、介入するハードウェア割込み出力が生じなければ次のROSアドレスである。ハードウェア割込みが生じれば、この新たに生成された形式的な次のROSアドレスは戻りアドレス・スタック70の最上部に置かれ、生成されたハードウェア割込みベクトル・アドレスがこれをRAR63における次のROSアドレスとして置換する。

テスト分岐操作の場所におけるように、マイクロ指令PUSHは主分岐操作と共に使用することができる。再び前に示したように、ハードウェア割込みはビットRDDT34を2進数1にセットすることにより禁止されなければならない。もしマイクロ指令PUSHが主分岐操作と関連して使用されるならば、主分岐マイクロ命令自体の(RoSアドレス履歴レジスタ66からの)アドレスである現在のROSアドレスプラス1は戻りアドレス・スタック70の最上部に置かれることになり、マイクロプログラムは主分岐ロジック57の出力により決定される形式的な次のアドレスに対して分岐することになる。

現在のマイクロ命令のROSアドレス・フィールド(13ビットのフィールドのビット0乃至3)において指定される如き定数による増分操作(マイクロ指令INCK)は、ROSアドレス履歴レジスタ66の現在の値を1だけ増分して次のマイクロサイクルに対してROSアドレス・レジスタに置かせる。この次のアドレスの生成に加えて、残る9ビット(RDDTビット4乃至12)は、その時のマイクロサイクルの間20ビット巾の内側バス38に対して8ビットの定数プラス充填数を生じるために使用される。

定数による増分マイクロ命令があるマイクロ命令のROSアドレス・フィールドにおいて指定される時、ROSアドレス履歴レジスタ66に含まれるその時のROSアドレスは増分機構64により増分され、その結果は、アドレス・マルチプレクサ1 60およびアドレス・マルチプレクサ2

10

20

30

40

50

62を介してROSアドレス・レジスタ63に対して戻される。もしハードウェア割込みが生じるならば、この新たに生成された次のアドレスは戻りマルチプレクサ61を介して戻りアドレス・スタック70の最上部に置かれ、ハードウェアが生成した割込みベクトル・アドレスは次のROSアドレスとして使用されるようにROSアドレス・レジスタ63に置かれることになる。副指令PUSHは増分操作と共に使用することができる。全ての副指令PUSHにおける如く、副指令PUSHが増分操作と関連して使用される時、ハードウェア割込みはRDDTビット34を2進数1にセット

することにより禁止されなければならない。もしマイクロ指令PUSHが増分操作と関連して使用されるならば、次のROSアドレスになることに加えて、1だけ増分された現在のROSアドレスが戻りアドレス・スタック70の最上部に置かれることになる。

マイクロ命令のROSアドレス・フィールドにおいて指定される如き増分操作 (INC) 副指令は、定数が内側バス38に対して生成されないことを除いて、定数による増分 (マイクロ指令INCK) のための前述のものと同じ操作を開始する。

戻り操作 (マイクロ指令POP) は、戻りアドレス・スタック70の最上部の内容を次のマイクロサイクルに対するROSアドレスとして使用されるようにアドレス・マルチプレクサ1 60とアドレス・マルチプレクサ2 62を介してROSアドレス・レジスタ63にロードさせる。更に、001 (16進数) のROSアドレスが、戻りアドレス・スタック70が1つの場所だけポップ操作される毎に、レジスタ74に対するスタックの底部にロードされる。ROSアドレス1による戻りアドレス・スタック70の底部へのローディングは、スタックの過剰ポップ操作の場合を検出するため用いられる。このスタックの過剰ポップ操作は、ROS場所1において開始するマイクロプログラムのエラー・シーケンスに対してマイクロプロセッサが誘導される結果となる。戻り操作 (マイクロ指令POP) はROSアドレス・フィールドのビット0乃至4により完全に指定されるため、ROSアドレス・フィールドのビット5乃至12は戻り操作の一部としては未使用である。

戻り操作がROSアドレス・フィールドのビット0乃至4において指定される時、介入するハードウェア割込みが生じなければ、ROSアドレス・レジスタ63は戻りアドレス・スタックの最上部の内容を受取る。ハードウェアの割込みが生じる場合には、戻り (即ちポップ) 操作は実際上バイパスもしくは打消されることになる。戻り操作の間にハードウェア割込みが生じる時のスタックのポップ操作のこの取消しは、戻りアドレス・スタック70の最上部からの戻りアドレスのポップ操作、および即時に同じマイクロサイクル以内にこれを戻りアドレス・スタック70の最上部に対してプッシュ・バックする操作と論理的に相当である。これは、使用される戻りアドレス・スタックがマイクロプログラムのサブルーチン呼出しおよ

びハードウェアの割込みの両操作のための戻りアドレスを含むことを許容する戻り操作の間にハードウェア割込みが生じる時の戻りアドレス・スタック70のポップ操作の取消またはバイパスである。

ハードウェア割込みルーチンのための戻りアドレスの格納と関連するスタックに対するプッシュ操作に、マイクロプログラム・サブルーチンから戻る時またはハードウェア割込みサービス・ルーチンの完了と同時にスタックにおいて実施されるポップ操作を取消させることにより、戻りアドレス・スタック77は反対方向に同時に移動すること、あるいは1つのマイクロサイクルの間に最初にポップ・アップ操作を次いでプッシュ・ダウンすることが可能である必要はない。反対方向のこの同時の運動が阻止される方法 (即ち、プッシュ操作がポップ操作を取消する方法) については、以下において第5図に関して更に詳細に説明することにする。以下の第4図の議論において判るように、ハードウェア割込みの発生と関連するプッシュ操作の発生による戻り操作と関連するスタックのポップ操作のこのような取消しは、ハードウェアの割込みルーチンがそのマイクロプログラムされた割込みサービス・ルーチンにおける最後のステップとして戻り操作を実施することになるため、マイクロプログラム内の制御のフローに悪影響を及ぼすことはない。

特別の制御フィールド (RDDT35乃至47) において符号化されるマイクロ指令PUSHは、望ましい実施態様におけるマイクロプロセッサ30内の結果が指定されないため、ROSアドレス・フィールド (RDDTバス0乃至12) 内で符号化される戻り操作と共に同じマイクロ命令において使用されてはならない。

前述の如く、ハードウェア割込みは固定されたROSアドレスに対して分岐を強制する。このROSアドレスは、レジスタ68からの入力として種々のエラー信号および割込み要求を有する優先順位回路網 (プリネット59) により決定される。ハードウェア割込みは、次のファームウェアが生じるROSアドレスを戻りアドレス・スタック70の最上部にプッシュさせる。次のROSアドレスが戻りマイクロ指令を介して戻りアドレス・スタックから生じたものであれば、戻りアドレス・スタック70のポップ操作が禁止される。マイクロ指令PUSH自体と関連する第1のプッシュ操作と、ハードウェア割込みのための戻りアドレスの保存と関連する第2のプッシュ操作であるスタックを二重にプッシュする要求を阻止するためマイクロ操作PUSHが行なわれる時は常にハードウェア割込みは禁止されなければならない。

ハードウェア割込み禁止フィールド (RDDTビット34) が2進数1である時、ハードウェア割込みは禁止され、非エラー条件のハードウェア割込み (メモリーリフレッシュおよびデータ要求と関連する如き) が禁止される (現在のマイクロ命令の実行および次のマイクロ命令の実行間に介入しないように阻止される)。プリネット59に対

するハードウェア・エラー条件入力(システム・クリア、存在しない資源へのアクセスの試み、アクセスの干渉、またはメモリー・パリティ・エラーの如き)はRDDTビット34の制御下ではなく、任意の2つのマイクロサイクル間に介入することができる。これらのエラー条件と関連するハードウェア割込みは禁止できないこと、従ってマイクロ指令PUSHを含むマイクロ命令の間に生じ得ることは、これらの禁止できないエラー条件と関連するハードウェア割込みサービス・ルーチンが終りに戻り操作を行なわないため、また従って戻りアドレス・スタック70の内容が有効であるかどうかには依存しないため、問題を生じることはない。RDDTビット34が2進数0である時は、全てのハードウェアを割込みが許容される。

第4図は、100により示される如きマイクロプログラムにおける主ルーチンを形成するファームウェアのマイクロ命令シーケンスである。主ルーチン100はそれ自体主なマイクロプログラム・ルーチンまたはマイクロサブプログラム・ルーチンであり、ある機能を実施するよう構成されたマイクロ命令の共通のシーケンスである。マイクロ命令のどの共通シーケンスに対する分岐でも実施することができることが望ましい。事例の目的のためには、第1のレベルのサブルーチンとして102で示されるものと第2のレベルのサブルーチンとして104で示されるものの2つのマイクロサブルーチンが示されている。これらのサブルーチンへの分岐操作はルーチンまたはサブルーチンにおけるどの場所でも生じることができる。更に、マイクロルーチン100およびその関連するマイクロサブルーチンのどれかの実行中に生じるどんなハードウェア割込みでも処理するためハードウェア割込みサービス・ルーチンを実施することができることが望ましい。事例の目的のためには、第1のハードウェア割込みサービス・ルーチンとしては106で、また第2のハードウェア割込みサービス・ルーチンとしては108で示される2つのハードウェア割込みサービス・ルーチンが示される。第4図においては、マイクロプログラムにおける矩形形状のブロックがマイクロ命令を示している。マイクロ命令の内容は、ROSデータ・レジスタ65に保有される如きROS24の出力により表示される。

第4図においては、第1のレベルのサブルーチンに対する飛越しを決定するマイクロ命令がマイクロ命令Pにおいて見出される。このマイクロ命令は、マイクロ命令Qに対する条件付きまたは無条件分岐を有するROSアドレス・フィールド、およびハードウェア割込みを禁止するためセットされるハードウェア割込みビットと共にマイクロ指令PUSHを表示する特別な制御フィールドとからなる。マイクロ命令Qに分岐し、戻りアドレスをプッシュし、ハードウェア割込みを禁止するこのファームウェアのマイクロ命令は、矩形形状ブロックPに示されている。このマイクロ命令がマイクロプログラムの順次実行中に読出された時、以降の諸操作が生じる。マイクロ命令の

ROSアドレス・フィールドは、分岐操作が実施されることを表示する。この特別な制御フィールドは、ROSアドレス履歴レジスタ66における現在のROSアドレスが増分機構64により増分されて、戻りマルチプレクサ61を介して戻りアドレス・スタック70の最上部にプッシュされるべきことを表示する。これはROSアドレス履歴レジスタ66の内容を取る増分機構64により構成されるが、このレジスタはこの時点において命令PのROSアドレスを保有し、これを1だけ増分して、その結果増分機構64の出力は命令(P+1)と関連するアドレスとなり、これを戻りマルチプレクサ61を介してレジスタ71である戻りアドレス・スタック70の最上部のレジスタに格納する。このように、この時点においては、レジスタ71はアドレス(P+1)を保有することになる。ROS24のアドレスPに常駐するこのマイクロ命令はまた、場所Pにおけるこのマイクロ命令の実行と場所Qにおけるマイクロ命令である第1のレベルのサブルーチンの最初の命令との間におけるハードウェア割込みの発生を禁止するようセットされたハードウェア割込みビットを有する。

場所Qにおいて示される如き分岐の発生に続いて、マイクロサブルーチンの通常のシーケンス動作が生じる。もし第1のレベルのサブルーチン102が矩形形状のブロックRに示される如き別のサブルーチンを要求するならば、もう1つのマイクロサブルーチンへの別の分岐が実行される。このようにして、命令Pにおいて生じるものと同ジステップは、第1のレベルのサブルーチン102が第2のレベルのサブルーチン104に対して分岐する時命令Rについても打倒する。こうして、命令Rにおいて、条件付きまたは無条件分岐がROS24のアドレスSにおいて開始する第2のレベルのサブルーチン104に対して実施される。命令Rが実行される時、ROSアドレス履歴レジスタ66における現在のROSアドレスが増分機構64により1だけ増分され、戻りアドレス・スタック70に対して転送されて最上部のレジスタ71に保持され、レジスタ71の内容はレジスタ72にプッシュ・ダウンされる。このように、マイクロプログラムの実行におけるこの時点においては、レジスタ71はアドレス(R+1)を含み、レジスタ72はアドレス(P+1)を含んでいる。アドレス(R+1)は第2のレベルのサブルーチンがその完了と同時に戻るべき第1のレベルのサブルーチン内の命令であり、またアドレス(P+1)は第1のレベルのサブルーチンがその完了と同時に戻るべき主ルーチン内のアドレスである。場所Rにおけるマイクロ命令のROSアドレス・フィールドは、マイクロプログラムを矩形ブロック(マイクロ命令)Sに対して分岐させることになる値を有する。

第4図に示されるように、第1のハードウェア割込みは矩形ブロックRと関連するマイクロ命令の実行中中断状態となる。この第1の割込みは、ハードウェア割込み禁止ビット(RDDTビット34)を2進数1と等しくなるよう

にセットすることにより禁止することができる形式のものであり、従ってこのハードウェア割込みはマイクロ命令Rに続くファームウェアの通常のフローを転向させることは許されない。従って、マイクロ命令Rの実行の完了と同時に、場所Sへの分岐は完了し、矩形ブロックSと関連するマイクロ命令が実行されることになる。もし第1の割込みがエラー割込み形式のものであったならば、ハードウェア割込みはマイクロ命令Rの完了と同時に生じることになり、マイクロプログラムの実行のフローはハードウェアのエラー割込みと関連する場所に対して誘導されることになる。

マイクロ命令Sの実行中、ROS24から取出されるべき次のマイクロ命令のROSアドレスが第4図の説明から判るように取得される。ファームウェアの実行を場所Vにおいて開始する第1のサービス・ルーチン106に対して誘導するハードウェア割込みの発生がなければ、通常の次のアドレスは(S+1)である。このアドレスは、ROSデータ・レジスタ65におけるROSアドレス・フィールド内のビットがテスト分岐ロジック56または主分岐ロジック57からのビットの付加を行なうかどうかにかかわらずアドレス・マルチプレクサ160を介して多重化されることになる飛越もしくは分岐操作のいずれかをも表示することができる、ROSデータ・レジスタ65に含まれるマイクロ命令SのROSアドレス・フィールドを使用することによって決定される。あるいはまた、もし増分操作がマイクロ命令SのROSアドレス・フィールド内に表示されているならば、ROSアドレス履歴レジスタ66の内容が増分機構64により1だけ増分されることになり、12ビットの出力がマルチプレクサ160を介して多重化されることになる。いずれにしても、アドレス・マルチプレクサ160の出力はアドレス(S+1)となる。

マイクロ命令Sは割込みを禁止せず、前述のようにハードウェア割込みが中断されているため、アドレス・マルチプレクサ160の出力は戻りマルチプレクサ61を介して多重化されて戻りアドレス・スタック70の最上部のレジスタ71に対してプッシュされ、アドレス(P+1)はレジスタ72からレジスタ73に対してプッシュされ、アドレス(R+1)はレジスタ71からレジスタ72に対してプッシュされる。同時に、アドレス・マルチプレクサ262はプリネット59の出力を介して多重化され、第1のハードウェア割込みと関連するアドレスがROSアドレス・レジスタ63に入力され、それによってROSアドレス・レジスタ63が矩形ブロックVと関連するアドレスを保有し、マイクロ命令がROS24から取出される。マイクロ命令Sの実行の完了と同時に、マイクロ命令VがROS24から取出されてROSデータ・レジスタ65において復号および実行のため使用できるようになる。

マイクロ命令V乃至Xを含む第1のハードウェア割込みサービス・ルーチン106の実行について記述する前に、もしマイクロ命令Rがハードウェア割込み禁止マイクロ

指令を含まなければ、その実行中、マイクロプログラムの適正な実行シーケンスを保存するため戻りアドレス・スタック70の二重プッシュ操作が必要とされる状態が生じる。即ち、増分機構64により出力される戻りマルチプレクサ61に対するアドレス(R+1)と、戻りマルチプレクサ61を介してアドレス・マルチプレクサ160により出力される次のアドレスSは両方共戻りアドレス・スタック70に対してプッシュされなければならない。二重プッシュ操作は単一のマイクロサイクルの実行中に可能でないため、ハードウェア割込みの禁止が必要となる。もしマイクロ命令Rの実行中に中断状態となる第1の割込みがハードウェア禁止ビット34のセットにより禁止可能でないハードウェアのエラー割込みであった場合、マイクロ命令Rが実行された後マイクロプロセッサはマイクロ命令Vに分岐することになり、第1のハードウェア割込みサービス・ルーチンがマイクロ命令Sの実行の介入なしに開始する。しかしこの場合は、第1のハードウェア・サービス106は、ハードウェア・エラー割込みと関連するサービス・ルーチンがポップ操作と共に終了しないという点で第4図に示されたものとは異なることになる。その代り、これらのシステムのいくつかの再初期化機能を行なった後に予め定めた点に分岐することにより終了する。これらのハードウェア・エラー割込みは、ポップ操作が割込みが行なわれたマイクロプログラムの実行を再開する意図を表示する故に、ポップ操作は行なわない。禁止可能でないこれらのハードウェア・エラーは、それからの回復を行なうことができる形式のものでなく、従って割込み点への制御の戻しは行なわない。次にマイクロ命令V乃至Xを含む第1のハードウェア・サービス・ルーチン106に戻ると、命令VおよびW、およびその間の全ての命令がその内部にハードウェア割込み禁止マイクロコード(即ち、ビット34は2進数1にセットされる)を有することが判る。あるハードウェア割込みサービス・ルーチンの最後の(POP)マイクロ命令を除く全てのマイクロ命令の間ハードウェア割込みの発生を禁止することにより、マイクロプログラムは、第2の即ち後続のハードウェア割込みの発生が第1のハードウェア割込みの処理に対する割込みを行わないことを保証する。このことは、1つのマイクロ命令により開始される、時間に依存するどのシーケンスも、開始および終了マイクロ命令間の時間を延長する一切の介入ハードウェア割込みを行なうことなく後続のマイクロ命令により完了することができることを保証する。これはまた、望ましい実施態様において、戻りアドレス・スタックが僅かに4つの階層しかないため、スタックの3のレベルはマイクロプログラムのサブルーチンの呼出しのため使用でき、また第4のレベルはハードウェア割込みが1階層以上にネストされる可能性なしにハードウェア割込みのため留保することができるという利点を有する。第2図におけるロジックが実際にハードウェア割込みのネステ

イングを許容すること、またもしスタックがその階層を超えることなく全てのハードウェア割込みが生じることを許容するに十分な階層を有し、許容されるマイクロプログラム・サブルーチン呼出しの最大ネesting深さであるならば、ハードウェア割込みサービス・ルーチンの間のこのような割込みの禁止は必要でなくなる(ハードウェア割込みがそれ自体に割込まないという前提に基づいて)ことが判るであろう。

第1のハードウェア割込みサービス・ルーチン106の実行中、マイクロ命令V乃至Wは割込みされことなく順次実行される。マイクロ命令Xは、マイクロ命令Wに禁止マイクロ指令が存在するので割込みなしにマイクロ命令Wの実行に続く。マイクロ命令V乃至WはINH(禁止)マイクロ指令に加えて生じた特定のハードウェア割込みのサービスと関連する諸操作を実施する他のマイクロ指令を含むことが理解されよう。同様に、マイクロ命令Xはポップマイクロ指令以外に他のマイクロ指令を含むこともできる。

マイクロ命令Xの実行中、ROSデータ・レジスタ65は、ROSアドレス・フィールドに符号化された戻り(POP)操作を有するマイクロ命令Xを有する。この戻り操作は、前述の如くアドレス・レジスタ71に(S+1)、レジスタ72に(R+1)、レジスタ73に(P+1)、レジスタ74にアドレス001(16進数)を保有する戻りアドレス・スタック70のポップ操作を生じる。マイクロ命令XにおけるPOPマイクロ指令は戻りアドレス・スタック70をして1つの場所だけポップさせる。マイクロ命令XのPOPマイクロ指令は、戻りアドレス・スタック70のレジスタ71からアドレス(S+1)をポップさせ、またレジスタ72の内容はレジスタ71にロードさせ、レジスタ73の内容はレジスタ72にロードさせ、レジスタ74の内容はレジスタ73にロードさせ、値001(16進数)をレジスタ74にロードさせる。アドレス(S+1)はアドレス・マルチプレクサ1 60およびアドレス・マルチプレクサ2 62を介してROSアドレス・レジスタ63にロードされ、ここでこのアドレスは後でROSデータ・レジスタ65にロードされるROS24からのマイクロ命令(S+1)を取出すため使用される。このようにして、マイクロ命令Xの実行に続いて、第2のレベルのサブルーチン104の実行がマイクロ命令(S+1)の実行と同時に再開する。

第2のレベルのサブルーチン104の実行は、マイクロ命令Tに至るまでは、命令毎にマイクロ命令(S+1)から続行する。マイクロ命令Xの実行後およびマイクロ命令Tの実行前は、戻りアドレス・スタック70はレジスタ71にアドレス(R+1)を、レジスタ72にアドレス(P+1)を、またレジスタ73およびレジスタ74にアドレス001(16進数)を保有する。

第4図は、第2の割込みが生じなければ戻りアドレス・スタック70の最上部からアドレス(R+1)をポップすることによりマイクロプログラムの実行をマイクロ命令

(R+1)に戻すことになる副指令POPを保有するマイクロ命令Tの実行中に、第2のハードウェア割込みが中断状態となることを示している。しかし、前述の如く、ハードウェア割込み指令と同時にマイクロ指令POPが生じると、戻りアドレス・スタック70のポップ操作はハードウェア割込みに対する応答と関連した暗黙のブッシュ操作により取消される。従って、戻りアドレス・スタック70はマイクロ命令Tの実行中ポップ操作およびブッシュ操作のいずれも行われず、スタックは、レジスタ71がアドレス(R+1)を、レジスタ72がアドレス(P+1)を、またレジスタ73およびレジスタ74がアドレス001(16進数)を保有するように変更されない状態を維持する。

マイクロ命令Tの実行中、ブリネット59はアドレスYがアドレス・マルチプレクサ2 62を介してROSアドレス・レジスタ63にロードされるように第2のハードウェア割込みサービス・ルーチンに対するベクトル化アドレスを生成する。マイクロプログラムの実行は、命令Yおよび命令Zからなる第2のハードウェア割込みサービス・ルーチン108における第1の命令であるマイクロ命令Yの実行と共に継続する。再び前述の如く、ハードウェア割込みサービス・ルーチン内の各命令は最後のマイクロ命令を除いてハードウェア割込みを禁止しなければならない。また、前述の如く、命令Yは禁止マイクロ指令に加えて他のマイクロ指令を保有し、またマイクロ命令Zは同様にマイクロ指令POPの以外に他のマイクロ指令を保有することができる。

マイクロ命令Zの実行中、ROSデータ・レジスタ65はROSアドレス・フィールドに戻り操作を保有することになる。この戻り(POP)マイクロ指令は、戻りアドレス・スタック70をして1つの場所だけポップ操作させ、これによりレジスタ71のアドレス(R+1)をアドレス・マルチプレクサ1 60およびアドレス・マルチプレクサ2 62を介して戻りマルチプレクサ61にロードさせる。更に、レジスタ72の内容はレジスタ71の内容を置換し、レジスタ73の内容はレジスタ73の内容を置換し、レジスタ74の内容はレジスタ72の内容を置換し、またアドレス001がレジスタ4にロードされる。このように、マイクロ命令Zの終りに、戻りアドレス・スタック70はレジスタ71においてアドレス(P+1)を、またレジスタ72、73、74においてアドレス001(16進数)を保有することになる。

マイクロ命令Zにおけるマイクロ指令POPによる戻りアドレス・スタック70のポップ操作は、マイクロプログラムの実行を第1のレベルのサブルーチン102におけるマイクロ命令(R+1)の実行と共に続行させることになる。第1のレベルのサブルーチン102は、第1のレベルのサブルーチンに分岐したマイクロ命令の後のマイクロ命令にマイクロプログラムの実行を戻すマイクロ指令POPを保有するマイクロ命令Uに至るまではそのマイクロ

命令の実行を継続する。マイクロ命令Uのマイクロ指令POPが戻りアドレス・スタック70を1つの場所だけポップ操作してアドレス(P+1)をアドレス・マルチプレクサ1 60およびアドレス・マルチプレクサ2 62を介してROSアドレス・レジスタ63にロードさせる。これは、ROS24における場所(P+1)にあるマイクロ命令をROSデータ・レジスタ65に対してロードさせてこれを実行させる。このように、マイクロプログラムの実行は、マイクロ命令(P+1)のROSアドレス・フィールドによりアドレス指定される次のマイクロ命令が後続する主ルーチン100におけるマイクロ命令(P+1)と共に継続することになる。

第4図に示された事例の以上の論述から判るように、あるサブルーチンに対する分岐と関連した各々の明白にマイクロコード化されたマイクロ指令PUSHは、ハードウェア割込みに対する応答と関連する各々の暗黙に行なわれたプッシュ操作における如く戻りアドレス・スタック70をして1つの場所だけプッシュ・ダウンさせる。上記の事例から、戻りアドレス・スタックが3つの階層のレベルまでプッシュされたに過ぎず、最初のプッシュは命令Pが第1のレベルのサブルーチンに対して分岐した時生じ、2番目のプッシュは命令Rが第2のレベルのサブルーチンに分岐した時生じ、3番目のプッシュは第1のハードウェア割込みサービス・ルーチンに対するベクトル化の準備中の命令Sの間に生じることが判る。

第4図の事例においては、戻りアドレス・スタック70のポップ操作は3回生じ、その最初は第1のハードウェア割込みサービス・ルーチンの完了時にマイクロ命令Xの間に生じ、2番目はマイクロ命令Zの実行中第2のハードウェア割込みサービス・ルーチンの終りに生じ、3番目は第1のレベルのサブルーチンが主ルーチンに戻る時マイクロ命令Uの実行中に生じる。通常、マイクロ命令Tにより第2のレベルのサブルーチンの終りに行なわれるポップ操作は、第2のハードウェア割込みサービス・ルーチンに対するベクトル化およびマイクロ命令Yの実行の開始と関連する暗黙のプッシュ操作により取消されたものである。

次に第5図に示された詳細な回路においては、前に用いたものと同じ照合番号がハードウェア機構の動作のために必要な詳細な回路に加えて示されている。第5図においては、種々の要素の入出力のあるものに付された小さな丸はそれぞれ反転入出力を表わしている。

第5図においては、戻りアドレス・スタック70は、Texas Instruments社により製造され参考のため本文に引用する同社の「The TTL Data Book for Design Engineers」第2版(1976年)に記載されたタイプSN74LS194なる12の4ビット両方向汎用シフト・レジスタからなる如くに示されている。第5図においては、戻りアドレス・スタック70における4つの12ビット・アドレスの最上位ビットを保持するシフト・レジスタ70Aと、4つの1

2ビット・アドレスの最下位の次のビットを保持するシフト・レジスタ70Kと、戻りアドレス・スタック70に記憶された4つの12ビット・アドレスの最下位ビットを保持するシフト・レジスタ70Lとのみが実際に示され、レジスタ70B乃至70Jは簡略化のため示さない。レジスタ71乃至74の各1ビットがシフト・レジスタ70A乃至70Lにそれぞれ格納されているため、第5図には戻りアドレスのレジスタ71乃至74が示されていることになる。12の4ビットのシフト・レジスタがプッシュ・ダウン・スタックとして使用される方法については以下に更に詳細に記述する。

第5図は、マイクロプロセッサ30を構成する種々の装置によって実行されるべきマイクロ命令を格納する読出し専用記憶装置24を示している。1つの操作サイクルにおいては、読出し専用記憶装置24はROSアドレス・レジスタ63の内容によりアドレス指定される。このため、このアドレスにより指定される場所の内容がROSデータ・レジスタ65に対して読込まれる。

第5図においては、ROSデータ・レジスタ65が第5図に関して記述されるべき問題となる3つの異なる場合を示す3つの個々のマイクロ命令を含む如く示されている。実際の操作においては、一時にこれらのマイクロ命令の1つのみがROSデータ・レジスタ65内に存在する。最上段のマイクロ命令はマイクロ命令Sへの飛越しを実施し、かつまたマイクロ指令PUSHを含む。このマイクロ命令は、第2のレベルのサブルーチン104への分岐を行なう第4図のマイクロ命令Rを示す。中段のマイクロ命令は、マイクロ命令内のハードウェア割込み(HIビット)の2進数1へのセットによりハードウェア割込みを禁止しない任意のマイクロ命令である。最下段のマイクロ命令は、ROSアドレス・フィールド内にマイクロ指令POPを保有し、かつマイクロ命令U、T、XまたはZの如き第4図の命令POPのどれかであり得るマイクロ命令である。このマイクロ命令POPは、サブルーチンまたはハードウェア割込みサービス・ルーチンからの戻りを行なうため使用される。

アクセスされるべき次のアドレスおよびマイクロプロセッサ30サブシステムを制御する次の指令を含むマイクロ命令ワードのフィールドは、第1図および第2図に示された機能素子に転送され復号される。各機能素子は、システムのタイミング操作の制御下でマイクロ命令ワードにより指定される所要の副指令を復号するための理論回路を含んでいる。

第1のレベルのサブルーチン102がマイクロ指令PUSHおよび禁止(INH)と共にSへの分岐マイクロ命令を実施することにより第2のレベルのサブルーチン104を呼出す時、以下のようなシーケンスが行なわれる。分岐マイクロ命令は、読出し専用記憶装置24から読出されてROSデータ・レジスタ65にロードされ、分岐は読出しに続くサイクルにおいて、ビット位置零に2進数1を有するRO

Sアドレス・フィールドから生じる。分岐マイクロ命令ワードはまた、戻りアドレスを戻りアドレス・スタック70に格納するマイクロ指令PUSHおよびハードウェア割込みを禁止するハードウェア割込み禁止のマイクロ指令とをそれぞれ有する。

更に、マイクロ指令JUMPは、ROSアドレス・フィールドのビット位置0から2進数1を受取りその出力において2進数0を生じるインバータ87によって復号され、その結果ドライバ60Aの出力可能(F)入力側と接続される信号JUMP-はその出力に対するドライバ60Aを可能状態にし、これによりアドレス・マルチプレクサ160の出力をマイクロ命令Sのアドレスを保有するROSアドレス・フィールドのビット1乃至12にさせる。ドライバ60Aは、アドレス・マルチプレクサ160を構成する唯一つのバス・ドライバのみが如何なる時もその出力を使用可能状態とするような3状態の出力を有するタイプのバス・ドライバである。アドレス・マルチプレクサ160は、第5図に示されるドライバ60Aおよび60B以上のものからなり、第5図に示されたもの以外の他の入力も有するが、第5図には本発明に関して有意義なものしか示さない。

アドレス・マルチプレクサ262のA入力側に現われるアドレス・マルチプレクサ160の出力は、アドレス・マルチプレクサ262の出力がROSアドレス・フィールドのビット1乃至12に現われるマイクロ命令Sのアドレスとなるように2進数1の状態にある選択入力(SEL)によって選択される。アドレス・マルチプレクサ262のこの出力は、命令Sが読出し専用記憶装置24でアドレス指定されるようにROSアドレス・レジスタ63にロードされる。

アドレス・マルチプレクサ262および戻りマルチプレクサ61の選択(SEL)入力に現われる選択入力信号LDRSAR+は、マルチプレクサ62および61の入力AおよびBの間を選択するため使用される。ハードウェア割込みブリネット・エンコーダ59の活動状態の出力(P)に現われる信号LDRSAR+は、もしエンコーダ59の出力Q1乃至Q8の1つが2進数0の状態にあるならば、2進数0の状態にある。もし入力I0乃至I10の内の1つが2進数0の状態にあり、またはもしエンコーダ59の使用可能入力(EN)に対して与えられる入力可能信号DISHIP+が2進数0の状態にあるならば、エンコーダ59の出力Q1乃至Q8の内の1つは2進数0の状態にあることになる。

望ましい実施態様においては、ハードウェア割込みブリネット・エンコーダ59は、Texas Instruments社により製造され同社の「The TTL Data Book for Design Engineers」第2版に記載された2つのカスケード接続されたSN74148タイプの8対3優先順位エンコーダからなるものである。エンコーダ59のI0入力は、ハードウェアのクリア状態において2進数0となる信号である信号CLEARX-を受取るように接続されている。他の入力信

号I1乃至I10は、ハードウェア割込み要求またはハードウェア・エラー条件を表示する種々の信号に接続される。これら入力信号I1乃至I10の内の1つが2進数0の状態にある時は常に、エンコーダ59は2進符号化された出力Q1乃至Q8上に、サービスを要求する最も高い優先順位のハードウェア割込みもしくはハードウェア・エラーと対応する信号を生じる。

ROSデータ・レジスタ65に示される第1のマイクロ命令はハードウェア割込み禁止フィールドに2進数1を保有するため、信号RDDT34+が2進数1となってANDゲート84を部分的に可能状態にする。信号CLEARX-は、クリア状態ではないためこれもまた2進数1となる。ハードウェア・エラー条件が存在しないため、ANDゲート84の第3の入力、信号STOPIT+もまた2進数1となり、従ってANDゲート84の出力である信号DISHIP+は2進数1となってエンコーダ59の出力を使用不能状態にすることにより信号LDRSAR+を2進数1にする。

第1のマイクロ命令のマイクロ指令PUSHは、NANDゲート83およびインバータ85と共にNORゲート80、81、82によって復号される。副指令PUSHは、2進数000のAサブフィールドと、2進数X0X0のBサブフィールド(但し、Xビットは「can't care」条件)と、2進数100のCサブフィールドを有することによりマイクロ命令ワードの特殊制御フィールドで指定される。AサブフィールドはNORゲート82に対してビット35乃至37を入力することにより復号され、その出力は副指令PUSHが指定されるならば2進数1となる。BサブフィールドはNORゲート81に対してビット39および41を入力することにより復号され、その出力は副指令PUSHが指定されると2進数1となる。Cサブフィールドは、インバータ85によりビット42を反転しかつNORゲート80に対する入力としてビット43、44と共にその出力を用いて復号される。NORゲート80の出力は、もし副指令PUSHが指定されると2進数1となる。もし副指令PUSHが特殊制御フィールドにおいて指定されるならば、NANDゲート83に対する3つの入力は2進数1となり、その出力である信号PUSHOP-を2進数0にする。

PUSH信号に対して、PUSHOP-がNANDゲート90に対する1つの入力において2進数0であり、これに対する他の入力である信号LDRSAR+が2進数1でハードウェア割込みが中断状態でも使用可能状態でもないことを表示する場合、NANDゲート90の出力である信号PUSHNT+は2進数1となって、マイクロ指令PUSHがROSデータ・レジスタ65内で復号化されて、戻りアドレス・スタック70に関するプッシュ操作が実施されるべきことを表示する。信号PUSHNT+はハードウェア割込みが中断状態でハードウェア割込みが可能状態になる場合にも2進数1となる。信号PUSHNT+はNANDゲート91の1つである。NANDゲート91に対する他の入力は、ORゲート86の出力である信号POPOP-を入力として受取るインバータ88から出力されるPOPOP-

P+であ。

ROSアドレス・フィールドのビット0乃至4は、マイクロ指令POPを復号するため使用されるORゲート86に対して入力される。もしビット0乃至4が2進数0であれば、ORゲート86の出力はポップ操作が実施されることを表示する2進数0となる。マイクロ命令JUMP Sのビット0が2進数1を含むため、ORゲート86の出力は2進数1となってインバータ88の出力を2進数0にさせる。

NANDゲート91の1つの入力において2進数0である信号POPOP+は、その出力である信号PUSPOP-を2進数1にさせて、プッシュ操作とポップ操作の両方が同時に要求されていないことを表示する。ANDゲート92の両方の入力が2進数1に等しい場合、その出力である信号PUSNOT+は2進数1となってプッシュ操作が戻りアドレス・スタック70に関して行なわれることを表示する。信号PUSN0+がORゲート94の1つの入力において2進数1である場合は、その出力の信号PUSNEN+は2進数1となってプッシュ操作が戻りアドレス・スタック70に関して行なわれることを表示する。

ポップ操作信号POPOP+はまたANDゲート93に対する1つの入力である。ANDゲート93に対する両方の入力が2進数0である場合、その出力である信号POPNOT+は2進数0となってポップ操作が戻りアドレス・スタック70に関して行なわれないことを表示する。クリア操作が行なわれていないため、インバータ89の出力である信号CLEARX+は2進数0となり、従ってORゲート95に対する両入力は2進数0となり、その出力である信号POPEN+を2進数0にさせてポップ操作が戻りアドレス・スタック70に関して行なわれないことを表示する。

信号PUSNEN+は、12の4ビット両方向のシフト・レジスタ70A乃至70Lの各々のシフト・モード入力S0と接続される。同様に、信号POPEN+は、12の4ビットの両方向のシフト・レジスタ70A乃至70Lのシフト・モード入力S1と接続される。シフト・モード入力S0とS1は、この入力S0とS1が2進数0と等しい時は、シフト・レジスタのクロック(C)に入力におけるクロック信号が2進数0の状態から2進数1の状態に変換する時に、シフト・レジスタのシフト動作又は並列ローディングが生じないように2進符号化されている。S0が2進数0でありS1が2進数1の状態である時、左方のシフト(POP)操作がシフト・レジスタがクロックされると実施される。S0が2進数1でありS1が2進数0である時、右方のシフト(PUSH)操作がシフト・レジスタがクロックされると行なわれる。S0およびS1の両方が2進数1である時、各シフト・レジスタ内の全ての4ビット位置の並列ローディングが、クロックが2進数0の状態から2進数1の状態に変換する時に行なわれる。12の4ビットの両方向のシフト・レジスタの各々のリセット(R)入力は、レジスタがシステムの作動中リセット(クリア)されないため、2進数1にセットされる。

ROSデータ・レジスタ65における現在のマイクロ命令の実行の後期において、また次のマイクロ命令のアドレスがROSアドレス・レジスタ63に対してロードされた後、シフト・レジスタ70A乃至70Lのクロック操作によって戻りアドレス・スタック70をクロックするのに用いられたクロック信号PSTROB+またはROSアドレス履歴レジスタ66のクロック操作のためにも使用される。従って、2進数0から2進数1の状態に変換する信号PSTROB+による戻りアドレス・スタック70に対する戻りアドレスのクロック動作と同時に、ROSアドレス・レジスタ63において見出される次のアドレスを用いて取出されるべき読出し専用記憶装置24における次の場所のアドレスは、次のマイクロ命令の実行中早期に現在のアドレスとして使用するためROSアドレス履歴レジスタ66にロードされることになる。

アドレスSのROSアドレス・レジスタ63からROSアドレス履歴レジスタ66に対するクロック操作に先立つ現在のマイクロ命令の実行中早期に、ROSアドレス履歴レジスタ66は場所Sに対する飛越しを指定するROSデータ・レジスタ65におけるマイクロ命令(即ち、第4図におけるマイクロ命令R)のアドレスを含んでいる。このため、マイクロ命令Rの実行中早期に、ROSアドレス履歴レジスタ66は現在のアドレスRを保有することになり、また増分機構64の出力は戻しマルチプレクサ61の入力Aにおいて現われるアドレス(R+1)となる。以上で明らかなように、マイクロ命令JUMP Sの間選択信号LDRSAR+は2進数1となって戻しマルチプレクサ61の入力AをそのQ出力に対して使用可能状態にさせる。このため、シフト・レジスタ70A乃至70Lの右方シフト直列入力側(RI)に現われる信号STKI00乃至STKI11は2進数アドレス(R+1)と対応することになる。このように、クロック信号PSTROB+がマイクロ命令JUMP Sの実行中遅く生じる時、アドレス(R+1)はシフト・レジスタ70A乃至70Lの右方へ1位置のシフトにより戻りアドレス・スタック70に対してプッシュされることになる。

割込みを禁止してマイクロ指令PUSHを実行する第1のマイクロ命令であるマイクロ命令JUMP Sに関して第5図のロジックの機能について記述したが、ROSデータ・レジスタ65に示された第2と第3のマイクロ命令の実行について次に論述する。第5図に示された第2と第3の両方のマイクロ命令は特殊制御フィールドにプッシュ操作を含まない。従って、これら両方の命令の実行中、マイクロ指令PUSHの復号のため使用されるNANDゲート83の出力は2進数1となってプッシュ操作がこのマイクロ命令内では復号されないことを表示する。

第2のマイクロ命令は、前述のように、そのROSアドレス・フィールドに「次に読み出されるべきマイクロ命令」のROM24内のアドレスを含む。従ってこの第2のマイクロ命令には正規順序のマイクロ命令の他サブルーチン分岐以外の分岐命令を含む。

10

20

30

40

50

次に第2のマイクロ命令については、ビット位置34におけるハードウェア割込みの禁止ビットが2進数0であってハードウェア割込みが許容されることを表示することが判る。この2進数0の状態はANDゲート84の中間の入力において信号RDDT34+として現われるその出力である信号DISHIP+を2進数0の状態にさせ、これによりハードウェア割込みプリネット・エンコード59に対する入力を可能にする。もし2進数0の状態のハードウェア割込み信号が存在せず、2進数0の状態のハードウェア・エラー信号が存在せず、クリア状態も存在しなければ、エンコード59の全ての入力I0乃至I10は2進数1の状態にあり、従ってエンコード59の出力Pにおける活動状態の出力信号である信号LSRSAR+が2進数1となる。

信号LSRSAR+が2進数1であり信号PUSHOP-が2進数1であってプッシュ操作が指定されないことを表示する場合は、NANDゲート90の出力である信号PUSHNT+は2進数0となり、マイクロ指令PUSHも存在せずあるいはハードウェア割込みの中断状態もないため、プッシュ操作が実施されないことを表示する。ANDゲート92の1つの入力が2進数0である場合は、その出力信号PUSNOT+はORゲート94の1つの入力において2進数0となる。もしクリア操作が実施されていなければ、ORゲート94の他の入力である信号CLEARX+は2進数0となる。従って、ORゲート94の出力である信号PUSNEN+は2進数0となる。

ポップ操作が第2のマイクロ命令のROSアドレス・フィールドにおいて指定されないため、インパータ88の出力である信号POPOP+は2進数0となってポップ操作が指定されなかったことを表示する。ANDゲート93の1つの入力における2進数0である信号POPOP+はその出力である信号POPNOT+を2進数0にさせる。ORゲート95に対する両入力が2進数0であれば、その出力である信号POPEN+は2進数0となる。シフト・レジスタ70A乃至70Lの入力における両方のシフト・モード信号S0およびS1は2進数0であるため、シフト・レジスタは、2進数0の状態から2進数1の状態に変換しつつある信号PSTROB+によりクロックされる時にシフトも並列ローディングのいずれも行なわず、従ってこのマイクロ命令の実行中プッシュ操作またはロード操作のいずれも行なわれることがない。

この第2のマイクロ命令の実行中、アドレス・マルチプレクサ2 62に対するA入力が、選択信号LDRSAR+が前述の如く2進数1であるため、そのQ出力側にゲートされる。従って、アドレス・マルチプレクサ1 60の出力はROSアドレス・レジスタ63に対してロードされる。現在のマイクロ命令の実行中遅く、戻りマルチプレクサ61の出力が2進数0の状態から2進数1の状態に変換する信号PSTROB+によりストローブされる時、現在の第2のマイクロ命令の次のアドレスを含むROSアドレス・レジスタ63の出力がROSアドレス履歴レジスタ66に対してロードされる。その後、次のマイクロ命令が現在のマイク

ロ命令として実行される時、ROSアドレス履歴レジスタ66はその現在のアドレスを保有することになる。

もしハードウェア割込みが2進数0であるビット34により可能状態にされているこの第2のマイクロ命令の実行ハードウェア割込みが生じるならば、エンコード59のP出力からの活動状態の出力信号であるLDRSAR+は2進数0となってハードウェア割込み、またはエラー条件、またはクリア状態が生じたことを表示し、アドレス・マルチプレクサ2 62の入力Bがその出力Qに対して可能状態にされる。アドレス・マルチプレクサ2 62に対する入力Bは8つの最上位ビットにおける8つの先行する2進数0と共にエンコード59からの2進数符号化ビットQ1乃至Q8からなり、その結果アドレス・マルチプレクサ2 62のQ出力は特定のハードウェア割込みまたはハードウェア・エラーまたはクリア状態を取扱うようにプログラムされたハードウェア割込みサービス・ルーチンの開始位置と対応する12ビットの2進アドレスとなる。

アドレス・マルチプレクサ2 62の出力はROSアドレス・レジスタ63に対して入力され、これから読出し専用記憶装置24に対して出力され、その結果ハードウェア割込みサービス・ルーチンの最初のワードの取出しおよび実行が可能となる。ハードウェア割込みサービス・ルーチンの最初のワードのアドレスはまた、ROSデータ・レジスタ65における第2のマイクロ命令の実行中に信号PSTAOB+がクロック（2進数0の状態から2進数1の状態へ変換）する時、ROSアドレス・レジスタ63からROSアドレス履歴レジスタ66へロードされる。このアドレスがROSアドレス・レジスタ63からROSアドレス履歴レジスタ66へクロックされると同時に、戻りマルチプレクサ61の出力が戻りアドレス・スタック70に対してプッシュされる。

この戻りマルチプレクサ61の出力は、この戻りマルチプレクサ61の入力Bに現われる形式的な次のマイクロ命令アドレスとなるアドレス・マルチプレクサ1 60の出力である。前述の如く、戻りマルチプレクサ61の入力Bは、エンコード59からの活動状態の出力が存在する時は常に選択信号LDRSAR+が2進数0であるため、その出力Qに置かれる。シフト・レジスタ70A乃至70Lは、ORゲート94からの信号PUSNEN+が2進数1でありかつORゲート95からの信号POPEN+が2進数0である故に、プッシュ操作（右方シフト）を行なう。ゲート90乃至95に対する入力において信号LDRSAR+が2進数0であり、信号CLEARX+が2進数0であり、信号PUSHOP-が2進数1であり、信号POPOP+が2進数0である故に、信号PUSNEN+は2進数1であり、信号POPEN+は2進数0となるが、このゲートの作用については前に述べた。

第5図におけるROSデータ・レジスタ65に示された第3のマイクロ命令の実行中、ポップ操作がROSアドレス・フィールドにおいて指示される如く実施される。前に示したポップ操作はORゲート86によって復号され、その出

力である信号POPOP-は2進数0となってポップ操作が実施されることを表示する。2進数0である信号POPOP-は、アドレス・マルチプレクサ2 62の入力Aがシフト・レジスタ70A乃至70Lの出力QAと対応する12ビットである信号STK000乃至STK011となるように、アドレス・マルチプレクサ1 60の内部のドライバ60Bの出力を可能状態にする。これら12の信号は、戻りアドレス・スタック70内部の最上位にあるアドレスと対応している。ドライバ60Aの出力は2進数1である信号JUMP-により使用不能状態にされる。このように、アドレス・マルチプレクサ1 60の出力は、第4図のレジスタ71の内容と対応する戻りアドレス・スタック70における最上位にあるアドレスとなる。

このPOPマイクロ命令の実行中は、ハードウェア割込みは生じない。アドレス・マルチプレクサ2 62の入力Aは、出力Qとして出力されるよう選択され、これにより読出し専用記憶装置24から取出されるべき次の命令のアドレスを指定するROSアドレス・レジスタ63にロードされる。ハードウェア割込みが生じないこのようなPOP命令の場合のように、読出し専用記憶装置24から読出される次の命令は、戻りアドレス・スタック70内の最上位の場所によりアドレス指定された命令となる。このPOPマイクロ命令の実行中にハードウェア割込みが生じなければ、エンコーダ59により出力される信号LDRSAR+は2進数1となり、その出力がいずれも活動状態でないことを表示する。プッシュ操作が特殊制御フィールドに指定されなかったため、NANDゲート83からの信号PUSHOP-は2進数1となり、従ってNANDゲート90の出力である信号PUSHNT+は2進数0となってプッシュ操作が行なわれないことを表示する。もしクリア操作が2進数0である信号CLEARX+により表示される如く生じなければ、ORゲート94の出力である信号PUSHEN+は2進数0となり、ORゲート95の出力である信号POPEN+は2進数1となって左方のシフト (POP) 操作がシフト・レジスタ70A乃至70Lにより実施されることを表示する。この状態は、信号PSTROB+が2進数0の状態から2進数1の状態に変換する時、1つの場所だけ戻りアドレス・スタック70をポップ操作させることになる。

シフト・レジスタ70A乃至70Lの左方のシフト操作が生じる時、12の4ビットシフト・レジスタの左方シフトの直列入力 (LI) におけるビットがシフト・レジスタに対してロードされる。第5図において判るように、シフト・レジスタ70A乃至70Kはこの入力に2進数0を保有し、シフト・レジスタ70Lはこの入力において2進数0を保有し、このため16進数値でロードされる第4図の戻りアドレス・スタック (レジスタ74) 内の最下位の場所が001として使用される結果となる。この001の値は、もしこの001の16進数がスタックの最上部からポップ・オフ操作される場合に生じるスタックの過剰ポップ操作の場合を取扱うようマイクロプログラムされたルーチンの読出

し専用記憶装置24内のアドレスとして使用される。

16進数001のこの同じROSアドレスは、クリア操作が実施される時戻りアドレス・スタック70のレジスタ71乃至74に対して同時にロードされる。これは、シフト・レジスタ70A乃至70KのPIA、PIB、PICおよびPID入力に2進数0を、またシフト・レジスタ70Lの対応する入力に2進数1を入力し、クリア操作の間2進数0の状態から2進数1の状態へ変換する信号PSTROB+によりクロックされる時、これらのレジスタのS0およびS1入力に2進数0に等しい信号PUSNEN+およびPOPEN+を入力して並列ローディングを行なうことにより実現される。これは、クリア操作の間CLEARX+が2進数1であり、これによりORゲート94と95の出力が2進数1に強制されてシフト・モード選択信号S0およびS1が共に2進数1に等しくなるようセットされることによって行なわれる。

第4図に関して前に述べたように、もし第5図に示された第3図のマイクロ命令の場合のようなポップ操作を含むマイクロ命令の実行中にハードウェア割込みが生じるならば、このハードウェア割込みと関連するプッシュ操作がこのマイクロ命令のポップ操作を取消し、戻りアドレス・スタックはプッシュ操作およびポップ操作のいずれも行なわれない。これが行なわれる方法について、第5図に関して次に記述しよう。

もしROSアドレス・フィールドにポップ操作を含む第3のマイクロ命令が2進数0のハードウェア割込み禁止ビットを含む場合は、POPマイクロ命令の実行中ハードウェア割込みが生じ得る。前に示したように、POPマイクロ指令を含むマイクロ命令の実行中、ドライバ60Bの出力が、アドレス・マルチプレクサ1 60の出力が戻りアドレス・スタック70の最上部に含まれる戻りアドレスとなるように使用可能状態にされる。アドレス・マルチプレクサ1 60の出力は、アドレス・マルチプレクサ2 62の入力Aおよび戻りマルチプレクサ61の入力Bに対して与えられる。ここで、もしハードウェア割込みが生じると、エンコーダ59からの信号LDRSAR+は2進数0でアドレス・マルチプレクサ2 62と戻りマルチプレクサ61の両方の入力Bを選択する。このように、アドレス・マルチプレクサ2 62の出力は、ハードウェア割込みサービス・ルーチンが位置されるエンコーダ59により決定される如きベクトル化アドレスとなる。戻りマルチプレクサ61の出力は、戻りアドレス・スタック70の最上部の出力であるドライバ60Bの出力となる。

このようにして、ROSアドレス・レジスタ63は中断状態の最も高い優先順位の割込みと関連するハードウェア割込みサービス・ルーチンの起動アドレスがロードされ、この場所は読出し専用記憶装置24から取出されてハードウェア割込みサービス・ルーチンの実行を開始することになる。信号LDRSAR+が2進数0であれば、NANDゲート90の出力である信号PUSHNT+は2進数1となって、プッシュ操作が戻りアドレス・スタック70に対して行なわれ

るべきことを表示する。POPマイクロ指令がROSアドレス・フィールドにおいて符号化されているため、ORゲート86の出力である信号POPOP-は2進数0となり、これによりインバータ88の出力である信号POPOP+Qを2進数1にさせてポップ操作が戻りアドレス・スタック70に対して行なわれるべきことを表示する。NANDゲート91の両入力が2進数1であれば、その出力である信号PUSPOP-は2進数0となってプッシュ操作およびポップ操作の両方が戻りアドレス・スタック70に対して同時に行なわれるべきことを表示することになる。

2進数0である信号PUSPOP-はANDゲート92と93を使用不能状態にし、信号PUSNOT+およびPOPNOT+を2進数0にさせてプッシュ操作およびポップ操作のいずれも行なわれなことを表示する。クリア操作が実施されていないため、信号CLEARX+は2進数0となり、従ってORゲート94と95の出力は2進数0となってシフト・レジスタ70A至70Lに対してシフト操作もロード操作も行なわれなことを表示する。従って、マイクロ命令内のPOPマイクロ指令はハードウェア割込みの暗黙のプッシュ操作を取消し、その結果戻りアドレス・スタック70のプッシュ操作、ポップ操作あるいは並列ローディングのいずれも生じることがない。

以上の論述から、NANDゲート91がプッシュ操作およびポップ操作の同時の発生を検出し、かつこの条件が生じる時戻りアドレス・スタック70の同時のプッシュ操作およびポップ操作を禁止するために使用されることが判るであろう。この条件は、POPマイクロ指令がマイクロ命令内に符号化されており、同時にハードウェア割込みが生じる時に生じ得るが、その組合せはスタックのいずれの方向の移動も阻止する。以上の論議から、最下位のレジスタがスタックの過剰ポップ操作の検出のため使用されるルーチンのアドレスによりポップ操作される毎にスタックの最下位のレジスタにおいてロードされる方法、およびスタック内の各レジスタが過剰ポップ・サービス・ルーチンのこのアドレスを含むようにクリア操作の間にスタックが初期化される方法もまた明らかになる。

望ましい実施態様について記述したが、当業者により他の変更例もまた容易に示唆されよう。例えば、もしハードウェア割込みがハードウェア割込みサービス・ルーチンの間禁止されなければ、本発明のスタック機構はハードウェア割込みのネストのため使用することができる。また、望ましい実施態様は4ビットのシフト・レジスタに関して記述したが、もしこれ以上のビットを含むレジスタが使用されれば、スタックのプッシュ・ダウン深さは拡張されたシフト・レジスタにおけるビット数と等しくなり、このためサブルーチンおよび割込みの比較的大きなネスティング深さを許容することになる。更に、米国特許第3,909,797号に記載される如きフリップフロップからなるレジスタからなるスタック機構は、同時のプッシュ操作およびポップ操作を行なう試みを検出するた

め使用される本発明のロジックがプッシュ操作およびポップ操作のいずれも行なわれなように使用されることを前提として、望ましい実施態様のシフト・レジスタの代りに使用することもできる。

サブルーチンに対するROSアドレス・フィールドは条件付きもしくは無条件の分岐のいずれでもあり得ることを留意されたい。このことは、次のアドレスが読出し専用記憶装置24からのマイクロ命令のROSアドレス・フィールドによって連続的に与えられるため、本文に記述したシステムにおいては特に有利である。それにも拘らず、この特質はシーケンス・カウンタが使用される如きアドレス指定方法においても等しく適用可能である。両方のアドレス指定方法は、もしサブルーチンへの分岐が行なわれなかつた場合、直ちに実行されたであろう次の命令のアドレスの格納を可能にするPUSHマイクロ指令の使用を必要とする。

本発明についてはその望ましい実施態様に関して示し記したが、当業者によれば、本発明の主旨および範囲から逸脱することなく上記および他の形態および詳細の変更が可能であることが理解されるであろう。

【図面の簡単な説明】

第1図は本発明を使用するデータ処理システムを示す全体ブロック図、第2図は本発明の原理を実施する第1図のマイクロプロセッサの制御領域の一部を示すブロック図、第3図は第1図のマイクロプロセッサのファームウェア・マイクロ命令ワード・フィールドを示す図、第4図は本発明に従って主ルーチンにおいて開始し、2つのサブルーチンおよび2つのハードウェア割込みサービス・ルーチンを続行するマイクロプログラムのマイクロ命令の実行を示すフロー図、および第5図は本発明による第2図の制御領域において使用される回路を示す理論的ブロック図である。

10……主記憶装置、12……主記憶装置モジュール0、14……主記憶装置モジュール1、16……主記憶装置モジュール2、

20……中央演算処理装置（CPU）、22……モニター・ロジック、

24……読出し専用記憶装置（ROS）、26……主要バス、

30……マイクロプロセッサ、32……データ操作領域、

34……メモリー管理装置（MMU）、36……制御領域、

37……プロセッサ・バス、38……内部バス、

40……入出力バス、42……入出力コントローラ、

44……周辺装置、50……テスト・フリップフロップ、

51……Fレジスタ、52……Fカウンタ、

53……ソフトウェア割込みレジスタ、

54……ソフトウェア割込みプリネット、55……アドレス生成部分、

56……テスト分岐ロジック、57……主分岐ロジック、

58……レジスタ、59……プリネット、

60……アドレス・マルチプレクサ1、61……戻りマルチ

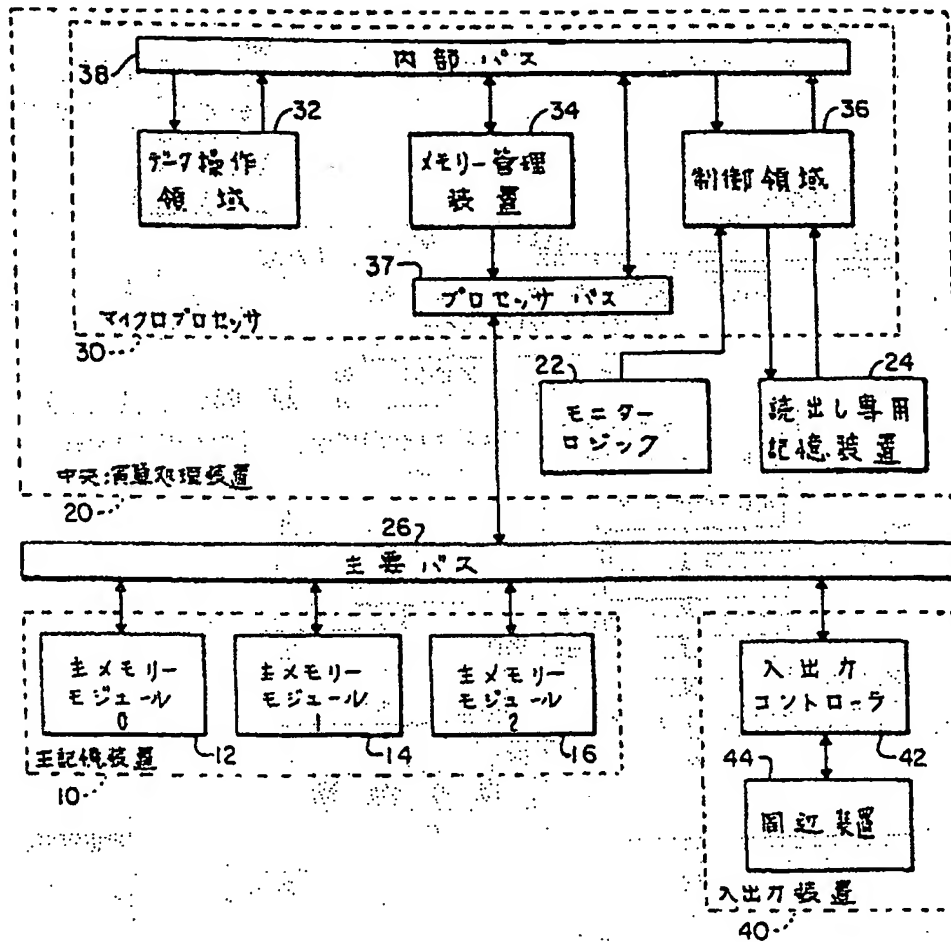
41

42

プレクサ、
 62……アドレス・マルチプレクサ2、63……ROSアドレ
 ス・レジスタ、
 64……増分機構、65……ROSデータ・レジスタ、
 65……ROSアドレス履歴レジスタ、70……戻りアドレス
 ・スタック、
 71~74……レジスタ、80~82……NORゲート、
 83……NANDゲート、84……ANDゲート、

* 85……インバータ、86……ORゲート、
 87~89……インバータ、90、91……NANDゲート、
 92、93……ANDゲート、94、95……ORゲート、
 100……主ルーチン、102……サブルーチン、
 104……サブルーチン、
 106……ハードウェア割込みサービス・ルーチン、
 108……第2のハードウェア割込みサービス・ルーチ
 * ン。

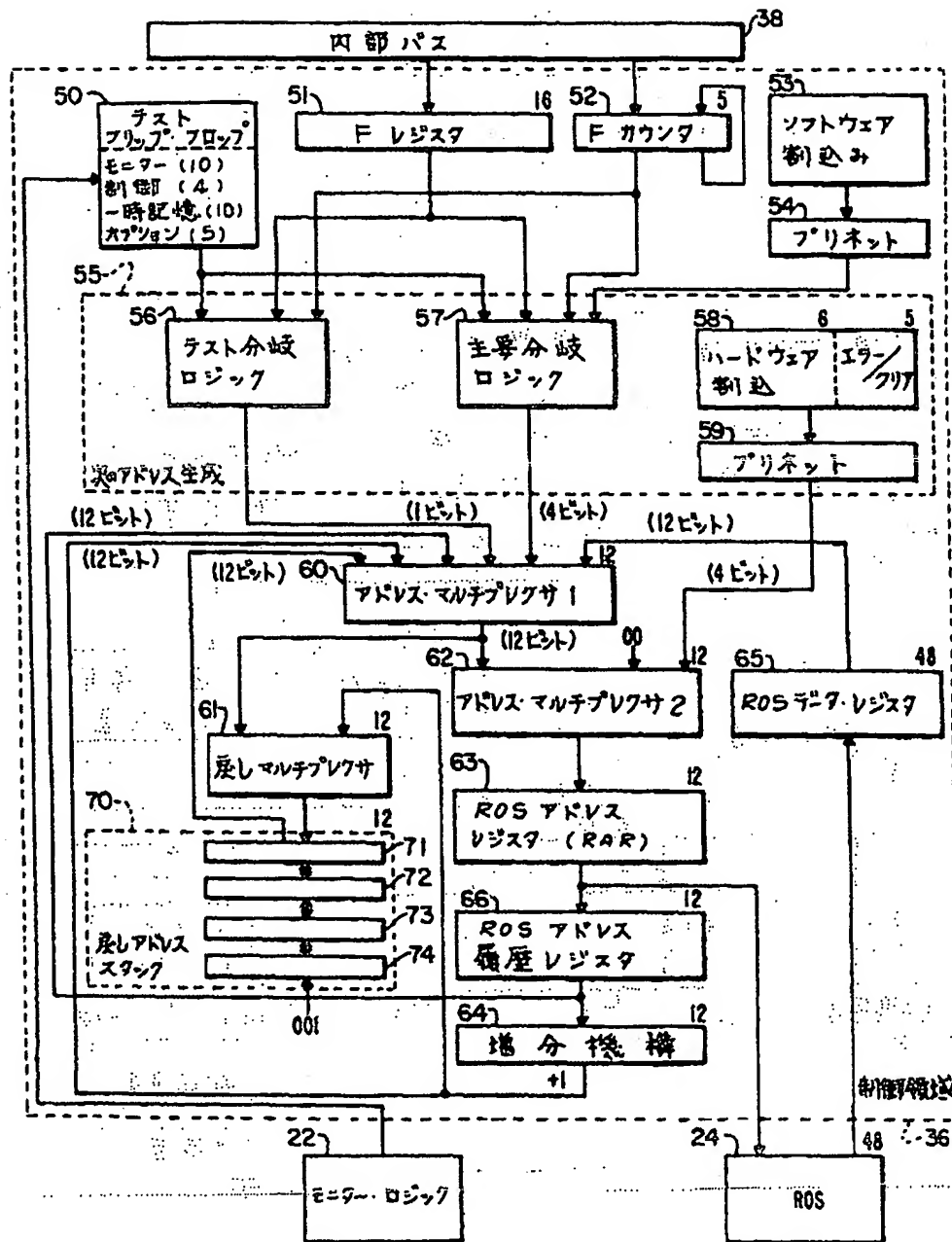
【第1図】



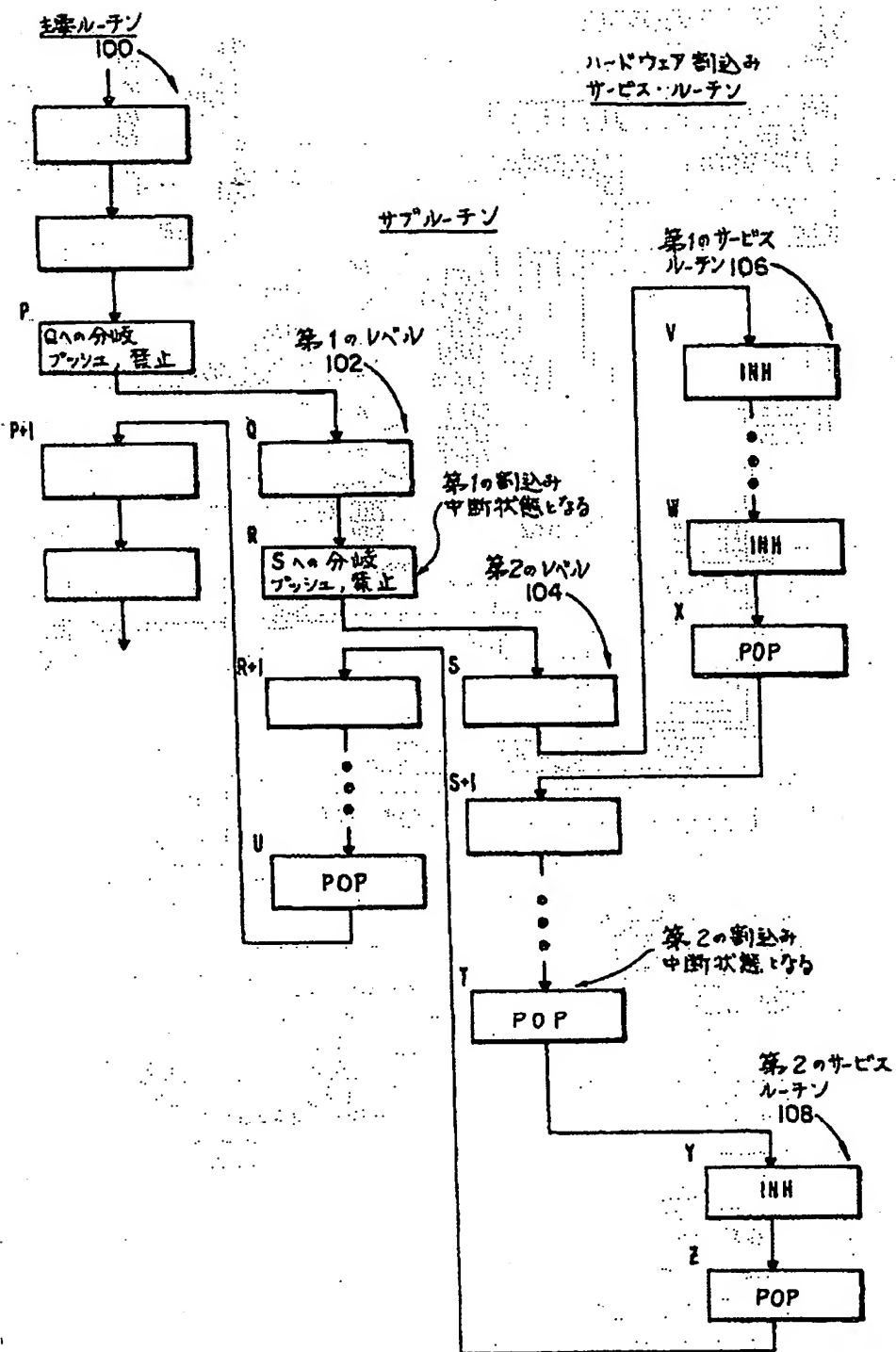
【第3図】

0	12 13	17 18	22 23 25 26	30 31 33 34	35 37 38	41 42	44 45 47
ROS アドレス	レジスタ・ファイル アドレス	ALU 制御装置	制御 バス	レジスタ変換 フィールド	NMU 割込み	A B C D	特殊制御 フィールド

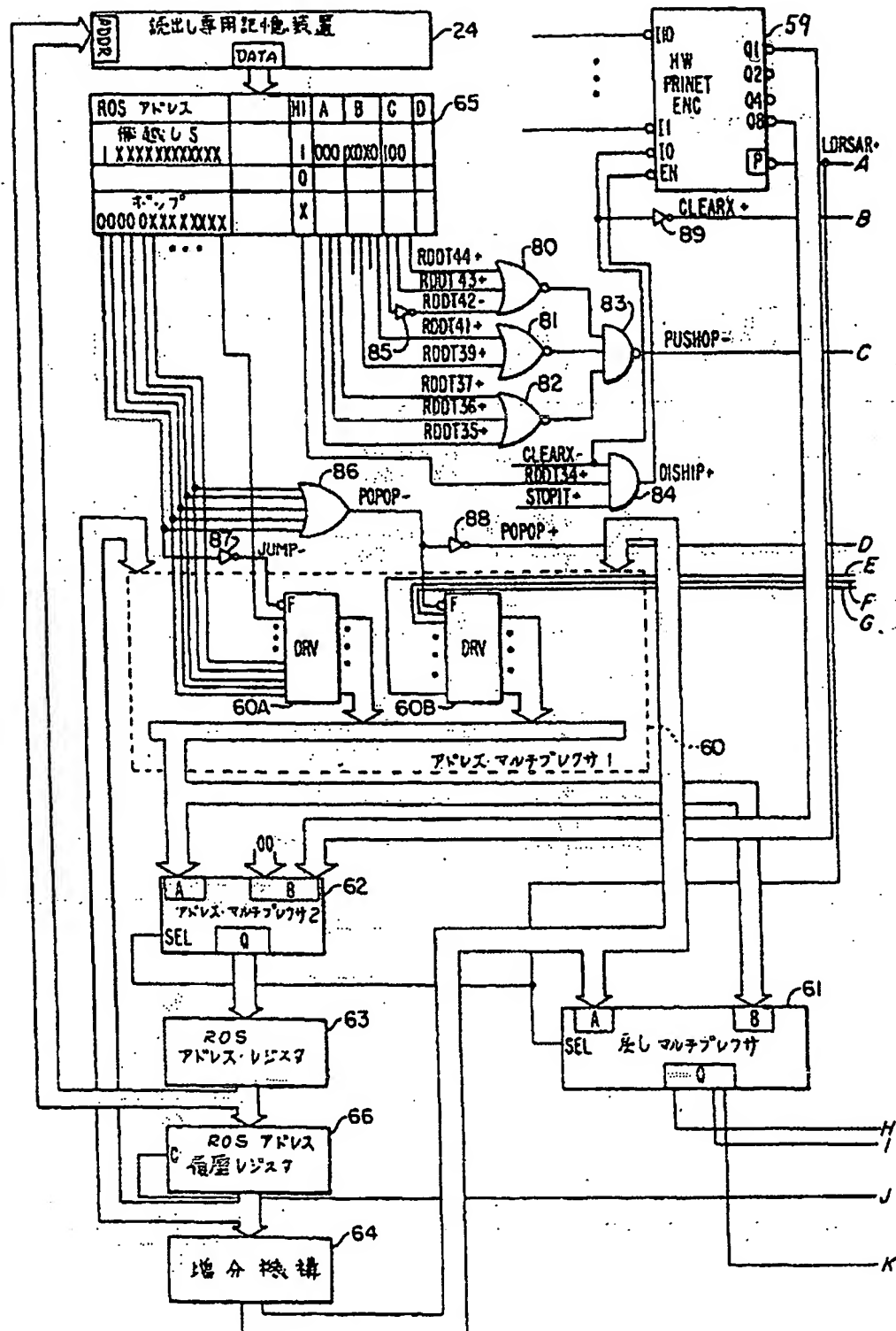
【第2図】



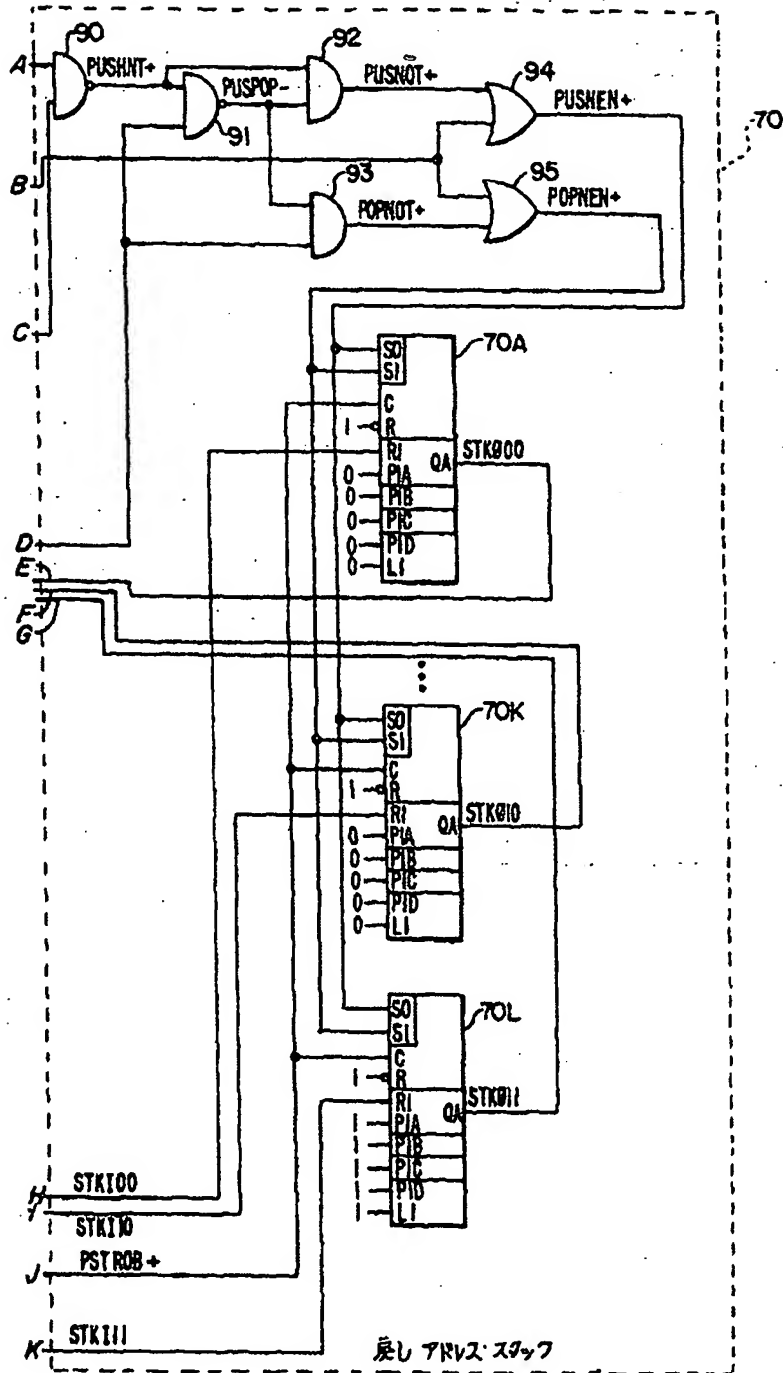
サブ⁹ルーチン



【第5図（2の1）】



【第5図(2の2)】



フロントページの続き

(72)発明者 ジョン・ジェイ・ブラッドレー
アメリカ合衆国マサチューセッツ州01701
フラミンガム・プレザント・ストリート
596

(56)参考文献 特開 昭55-52152 (J P, A)
特開 昭50-128429 (J P, A)
特開 昭54-106142 (J P, A)
特公 昭56-4943 (J P, B 2)